FIG. 1

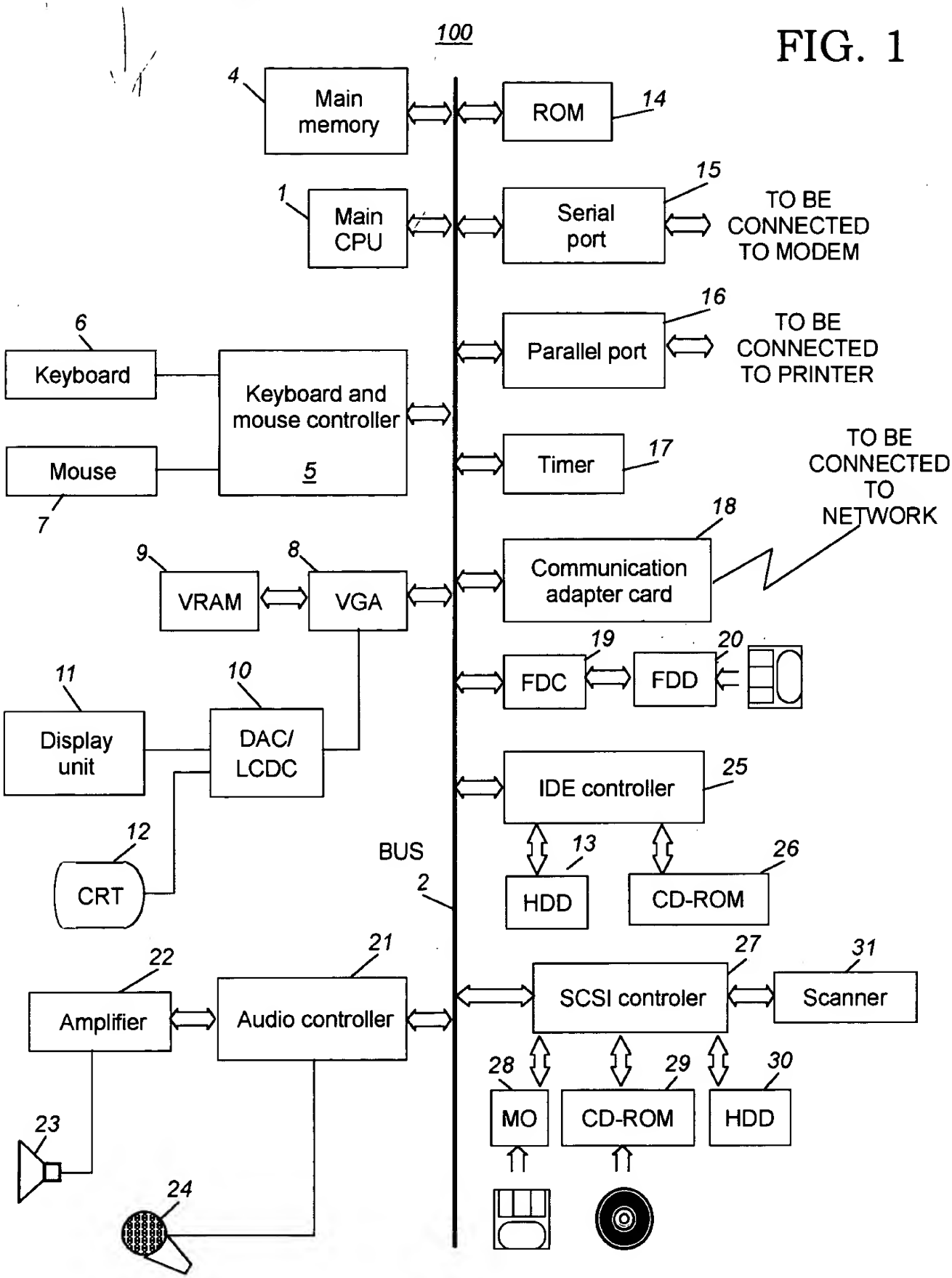

100
4
Main memory
ROM
14
1
Main CPU
15
Serial port
TO BE CONNECTED TO MODEM
16
Parallel port
TO BE CONNECTED TO PRINTER
6
Keyboard
Keyboard and mouse controller
5
Mouse
7
Timer
17
TO BE CONNECTED TO NETWORK
18
Communication adapter card
9
VRAM
8
VGA
19
FDC
20
FDD
11
Display unit
10
DAC/ LCDC
25
IDE controller
12
CRT
BUS
2
13
HDD
26
CD-ROM
22
Amplifier
21
Audio controller
27
SCSI controler
31
Scanner
28
MO
29
CD-ROM
30
HDD
23
24

FIG. 2A

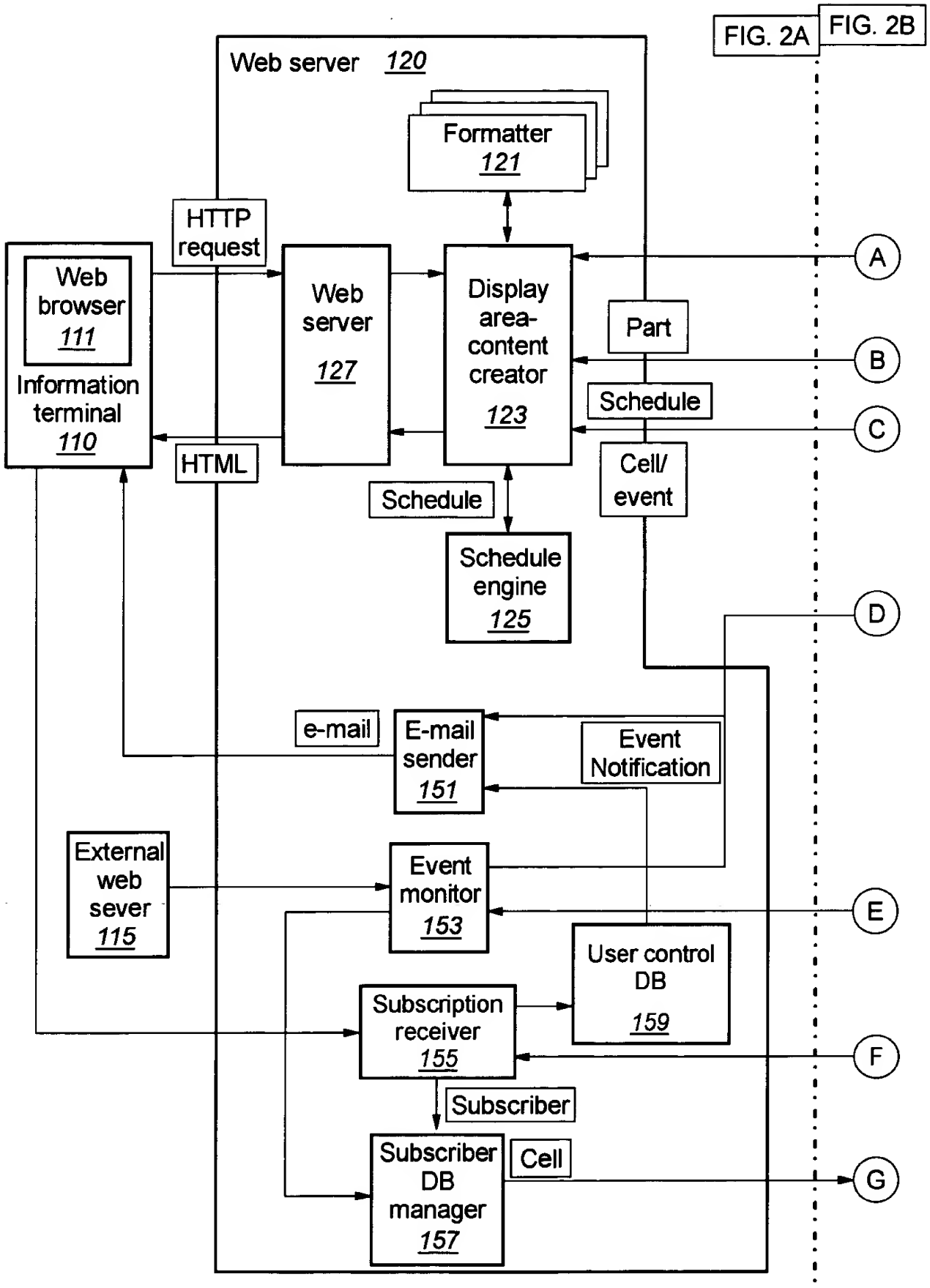


FIG. 2A
FIG. 2B
Web server 120
Formatter 121
HTTP request
Web browser 111
Information terminal 110
Web server 127
Display area-content creator 123
Part
Schedule
Cell/ event
HTML
Schedule
Schedule engine 125
A
B
C
D
E
F
G
e-mail
E-mail sender 151
Event Notification
External web sever 115
Event monitor 153
User control DB 159
Subscription receiver 155
Subscriber
Subscriber DB manager 157
Cell

FIG. 2B

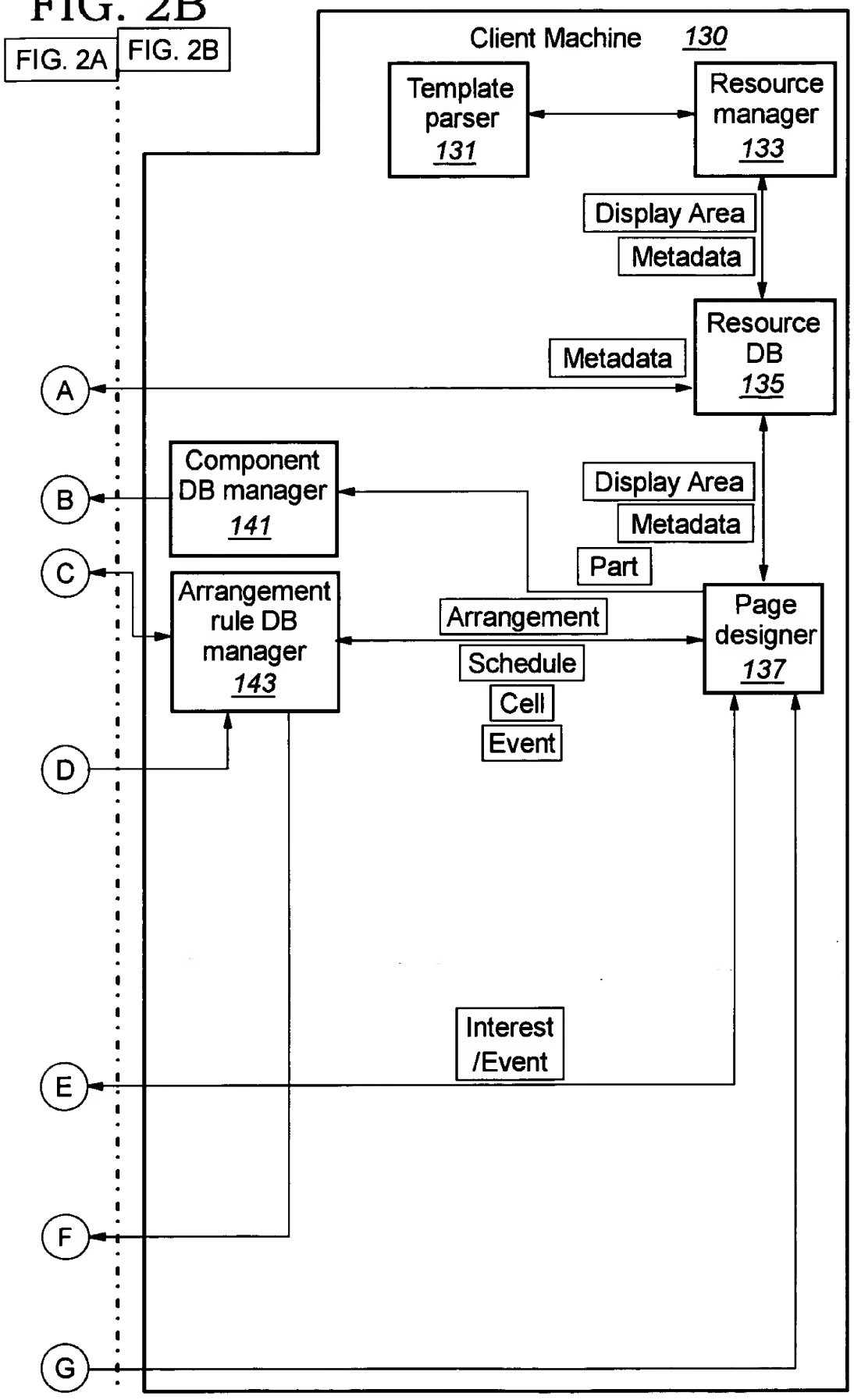


FIG. 2A
FIG. 2B
Client Machine 130
Template parser 131
Resource manager 133
Display Area
Metadata
Resource DB 135
Metadata
A
Component DB manager 141
B
Display Area
Metadata
Part
C
Arrangement rule DB manager 143
Arrangement
Schedule
Cell
Event
Page designer 137
D
Interest /Event
E
F
G

FIG. 3

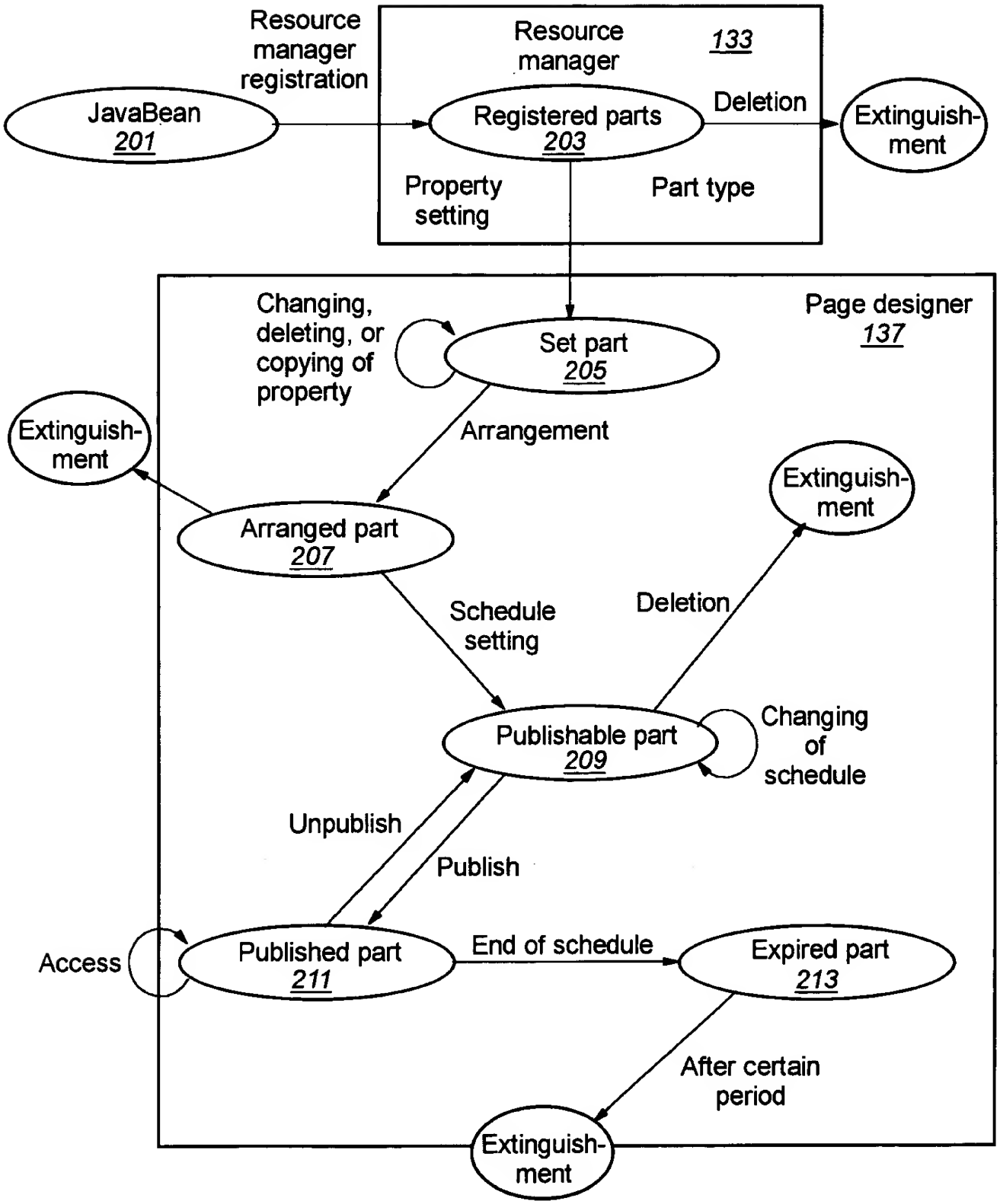

Resource manager registration
Resource manager
133
JavaBean
201
Registered parts
203
Deletion
Extinguish-ment
Property setting
Part type
Page designer
137
Changing, deleting, or copying of property
Set part
205
Extinguish-ment
Arrangement
Extinguish-ment
Arranged part
207
Schedule setting
Deletion
Changing of schedule
Publishable part
209
Unpublish
Publish
Access
Published part
211
End of schedule
Expired part
213
After certain period
Extinguish-ment

FIG. 4

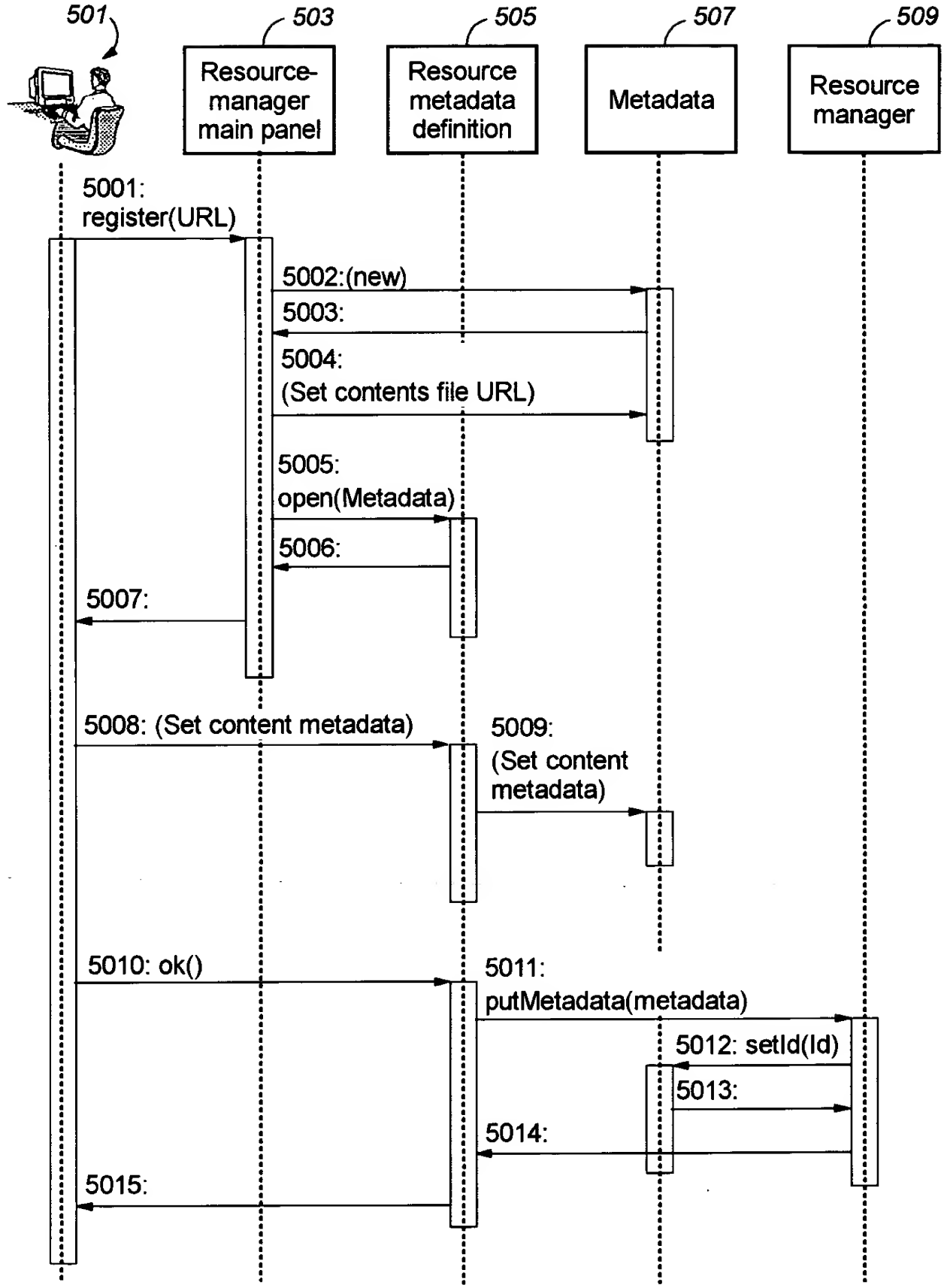

Content creator
501
503
505
507
509
Resource-manager main panel
Resource metadata definition
Metadata
Resource manager
5001: register(URL)
5002:(new)
5003:
5004: (Set contents file URL)
5005: open(Metadata)
5006:
5007:
5008: (Set content metadata)
5009: (Set content metadata)
5010: ok()
5011: putMetadata(metadata)
5012: setId(Id)
5013:
5014:
5015:

FIG. 5

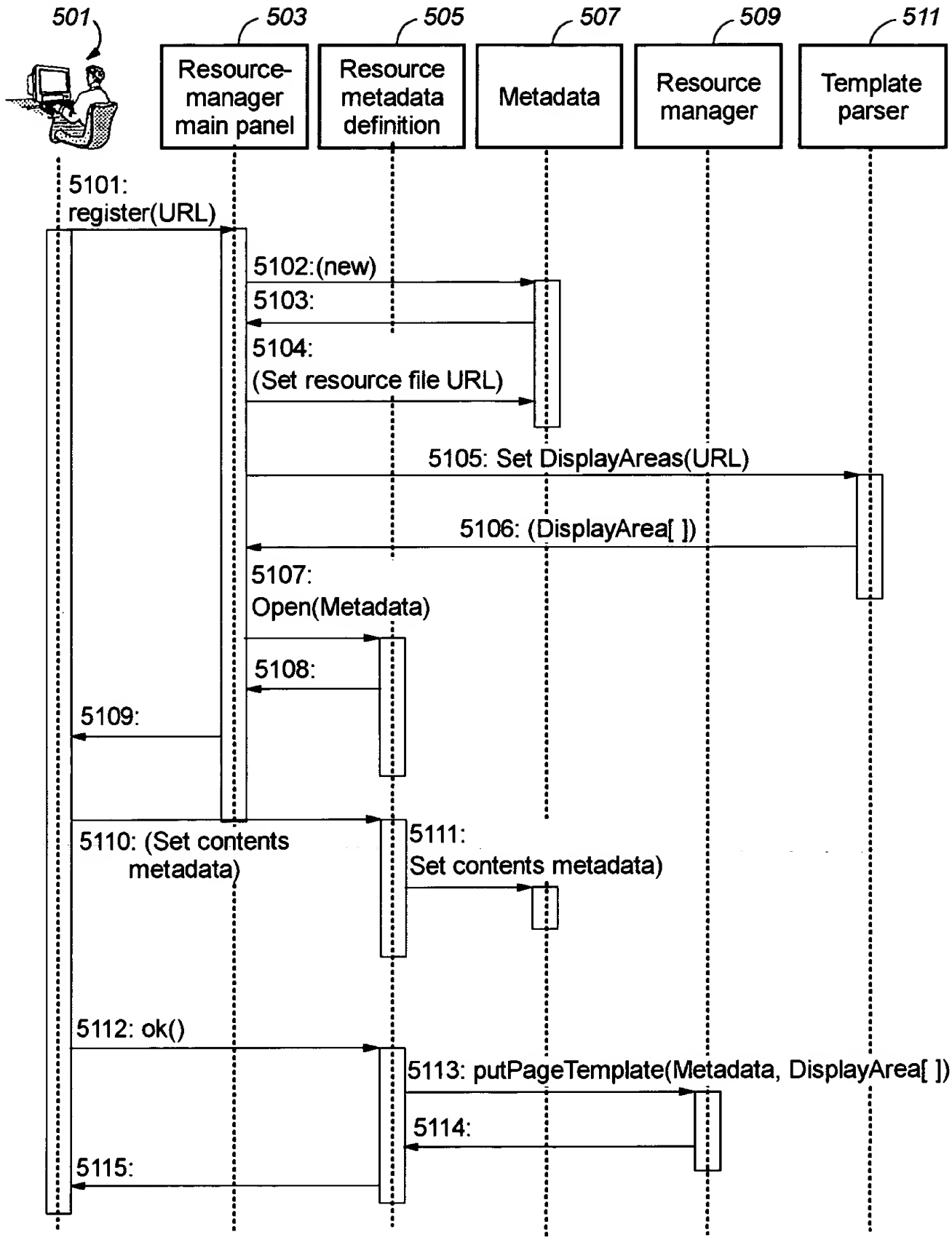

Content creator
501
503
505
507
509
511
Resource-manager main panel
Resource metadata definition
Metadata
Resource manager
Template parser
5101: register(URL)
5102:(new)
5103:
5104: (Set resource file URL)
5105: Set DisplayAreas(URL)
5106: (DisplayArea[ ])
5107: Open(Metadata)
5108:
5109:
5110: (Set contents metadata)
5111: Set contents metadata)
5112: ok()
5113: putPageTemplate(Metadata, DisplayArea[ ])
5114:
5115:

FIG. 6

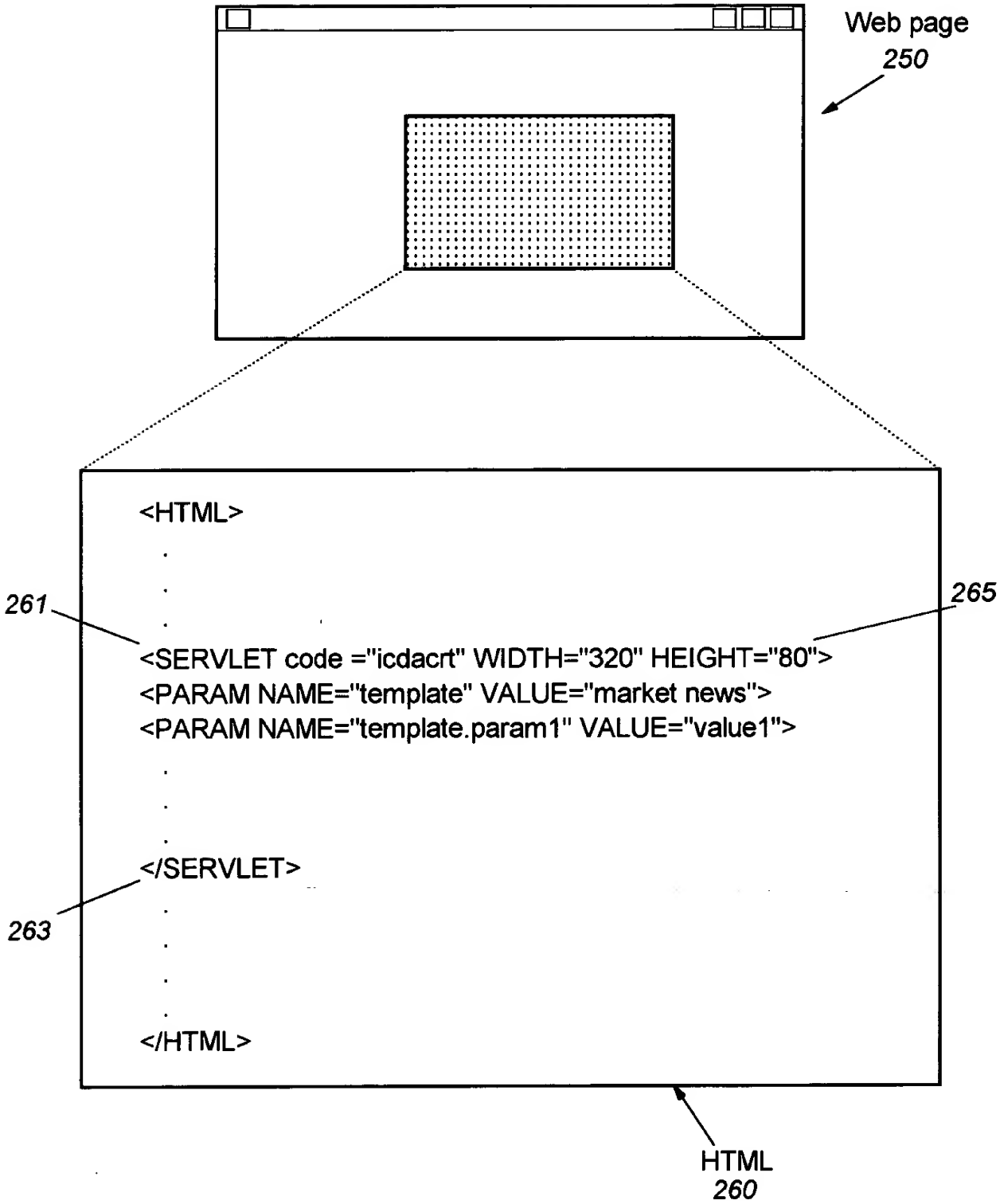
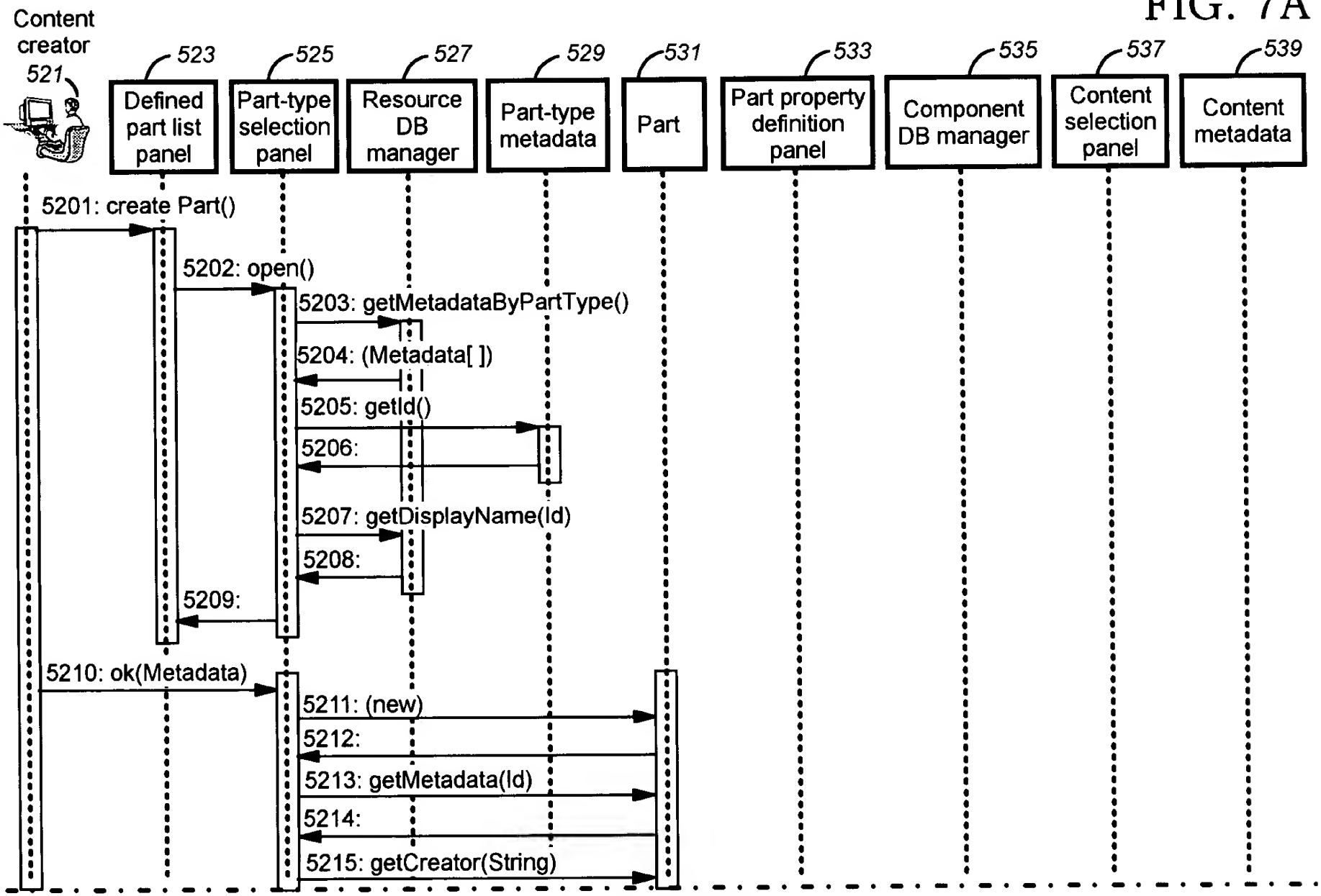

Web page
250
<HTML>
261
265
<SERVLET code ="icdacrt" WIDTH="320" HEIGHT="80">
<PARAM NAME="template" VALUE="market news">
<PARAM NAME="template.param1" VALUE="value1">
</SERVLET>
263
</HTML>
HTML
260

FIG. 7A



(To FIG. 7B)

FIG. 7B

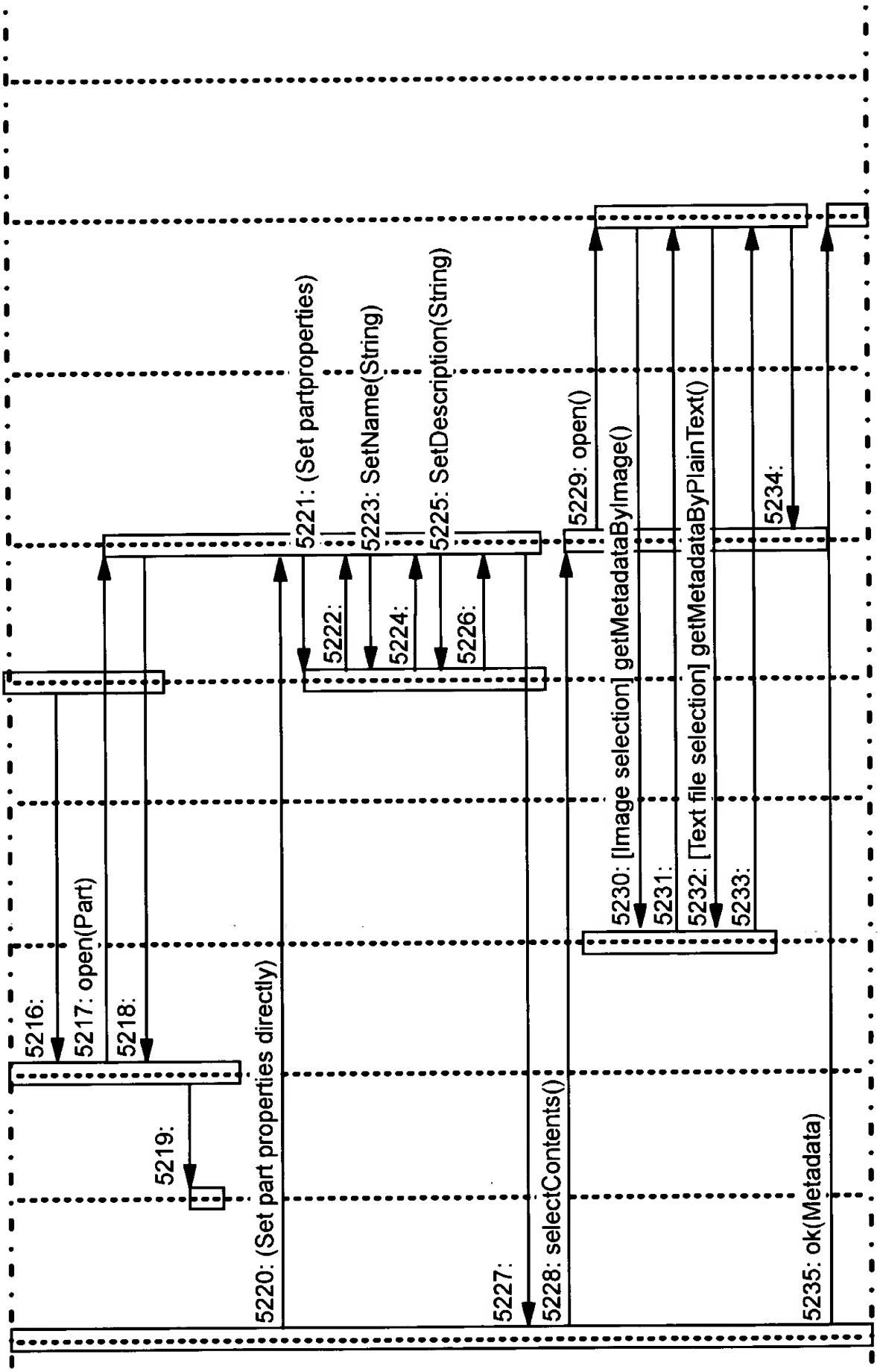

(From FIG. 7A)
5216:
5217: open(Part)
5218:
5219:
5220: (Set part properties directly)
5221: (Set partproperties)
5222:
5223: SetName(String)
5224:
5225: SetDescription(String)
5226:
5227:
5228: selectContents()
5229: open()
5230: [Image selection] getMetadataByImage()
5231:
5232: [Text file selection] getMetadataByPlainText()
5233:
5234:
5235: ok(Metadata)
(To FIG. 8)

FIG. 8

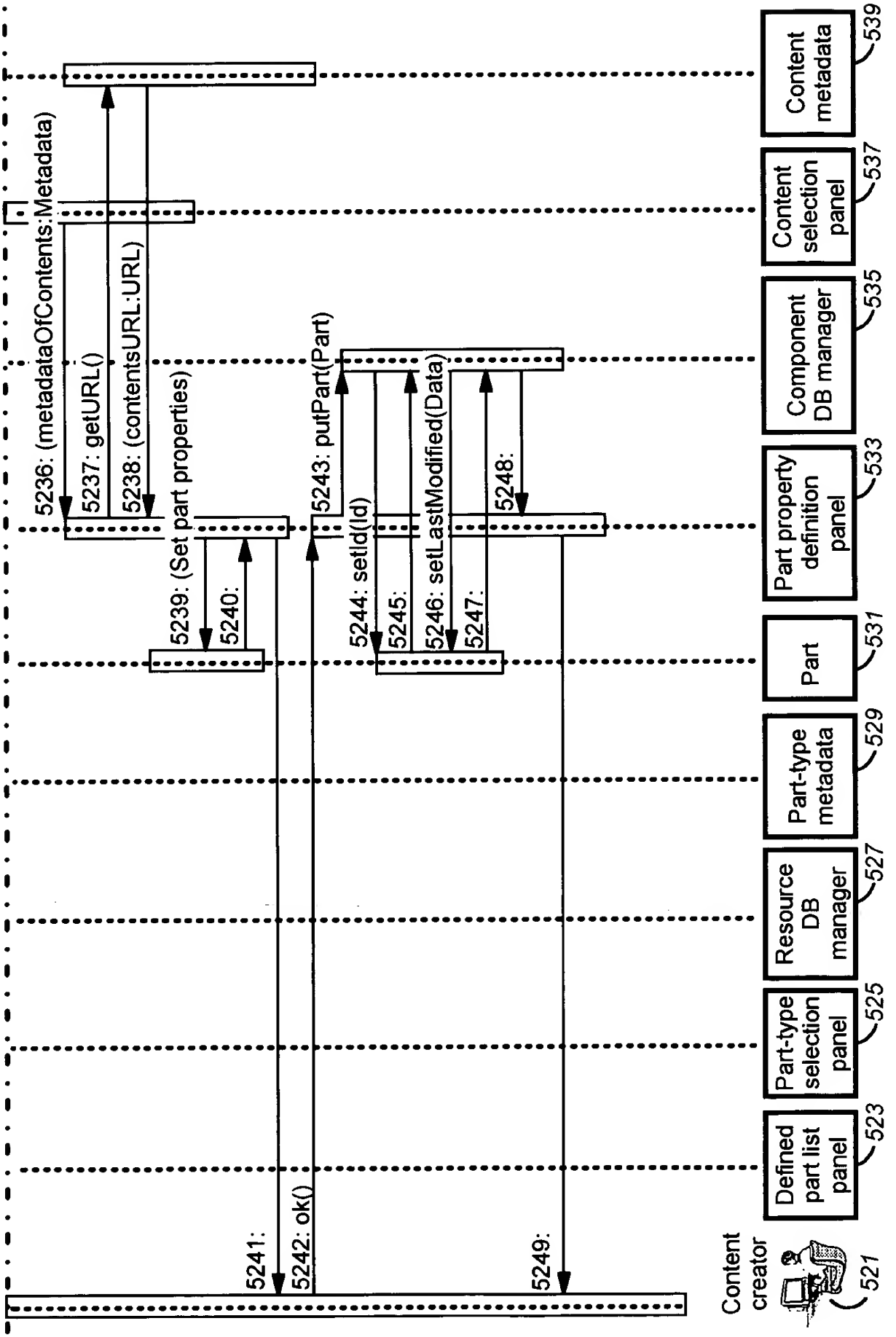
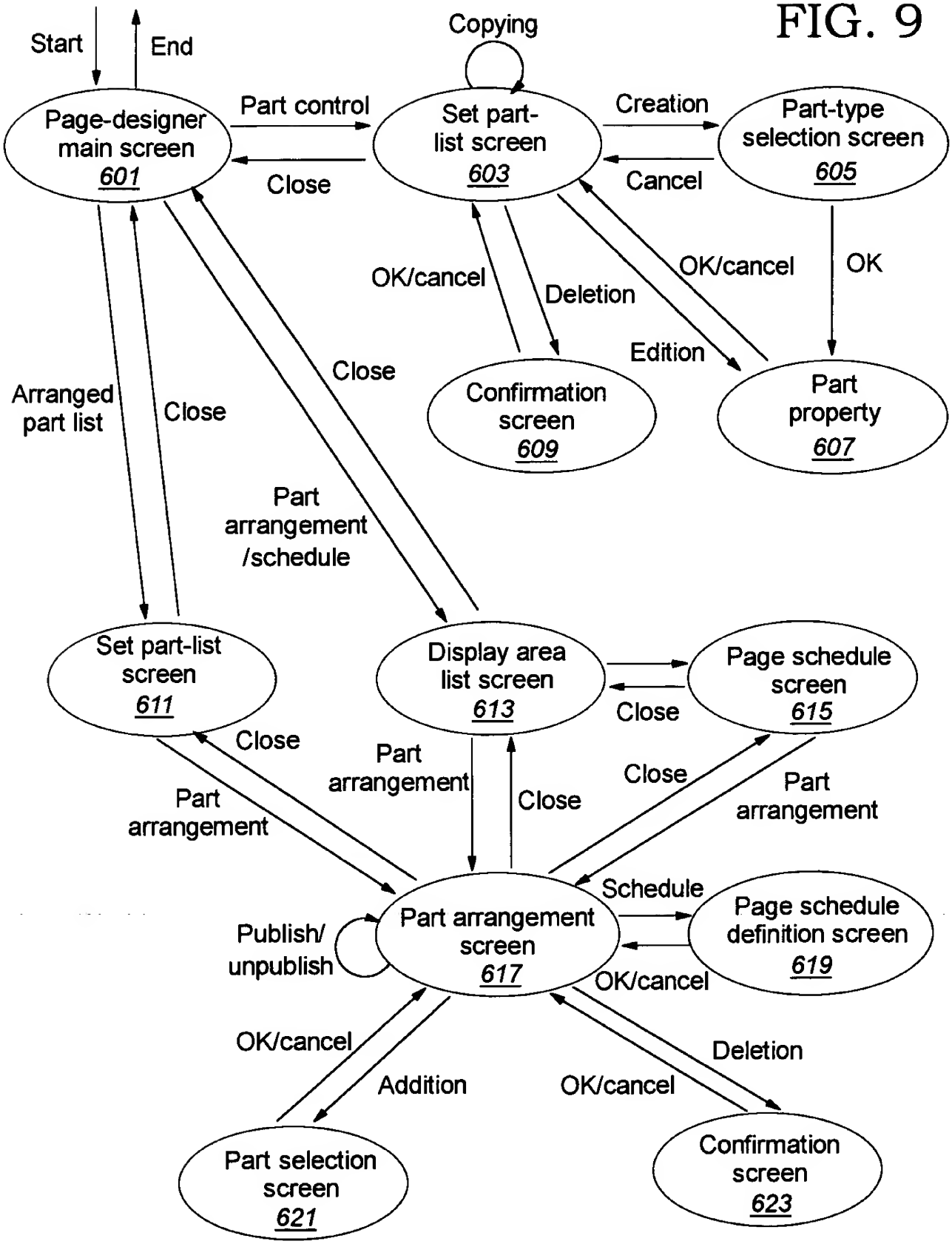
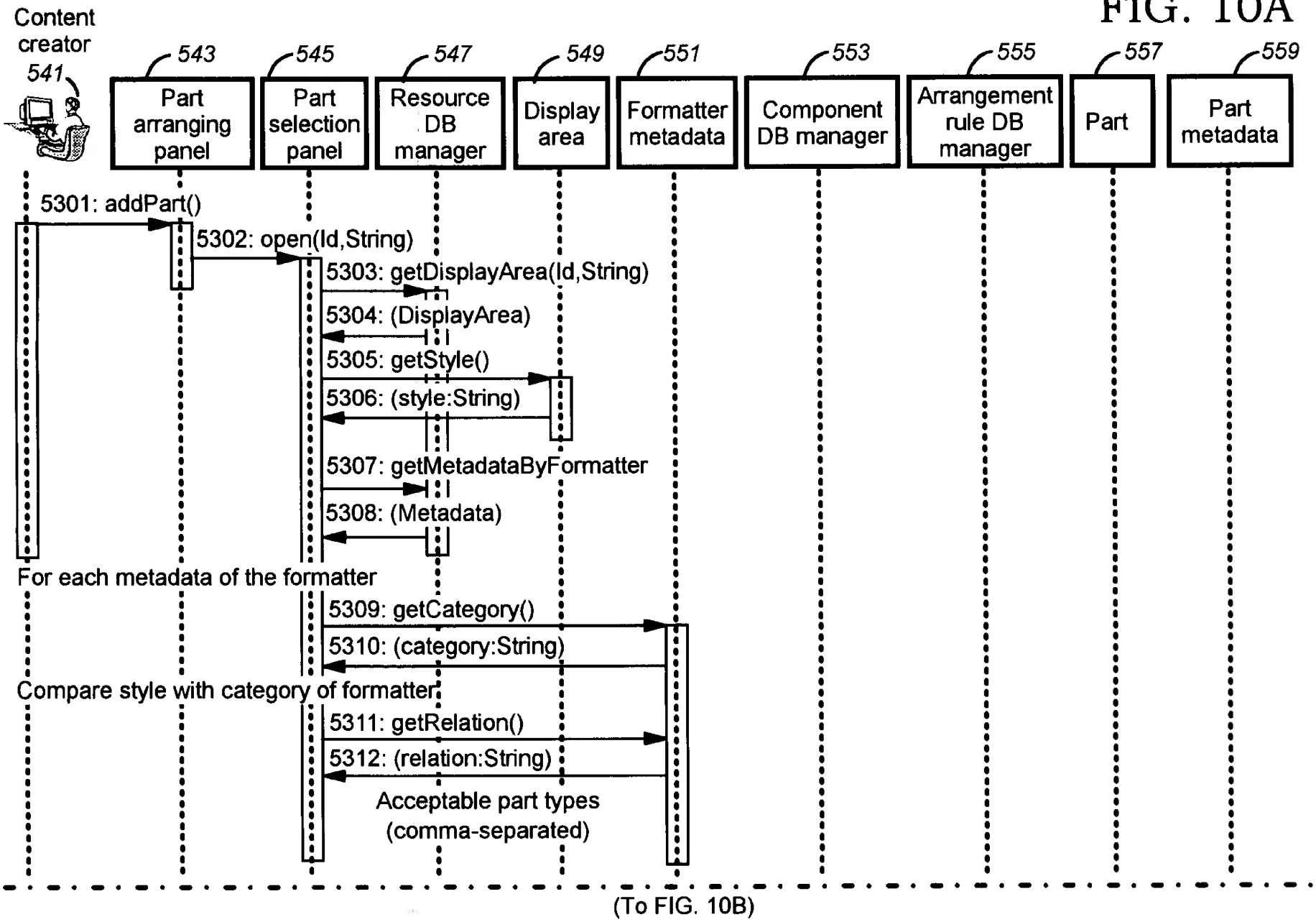

(From FIG. 7B)
Content creator
521
Defined part list panel
523
Part-type selection panel
525
Resource DB manager
527
Part-type metadata
529
Part
531
Part property definition panel
533
Component DB manager
535
Content selection panel
537
Content metadata
539
5236: (metadataOfContents:Metadata)
5237: getURL()
5238: (contentsURL:URL)
5239: (Set part properties)
5240:
5241:
5242: ok()
5243: putPart(Part)
5244: setId(Id)
5245:
5246: setLastModified(Data)
5247:
5248:
5249:

FIG. 9



08542006 - 07/14/06

FIG. 10A



(To FIG. 10B)

FIG. 10B

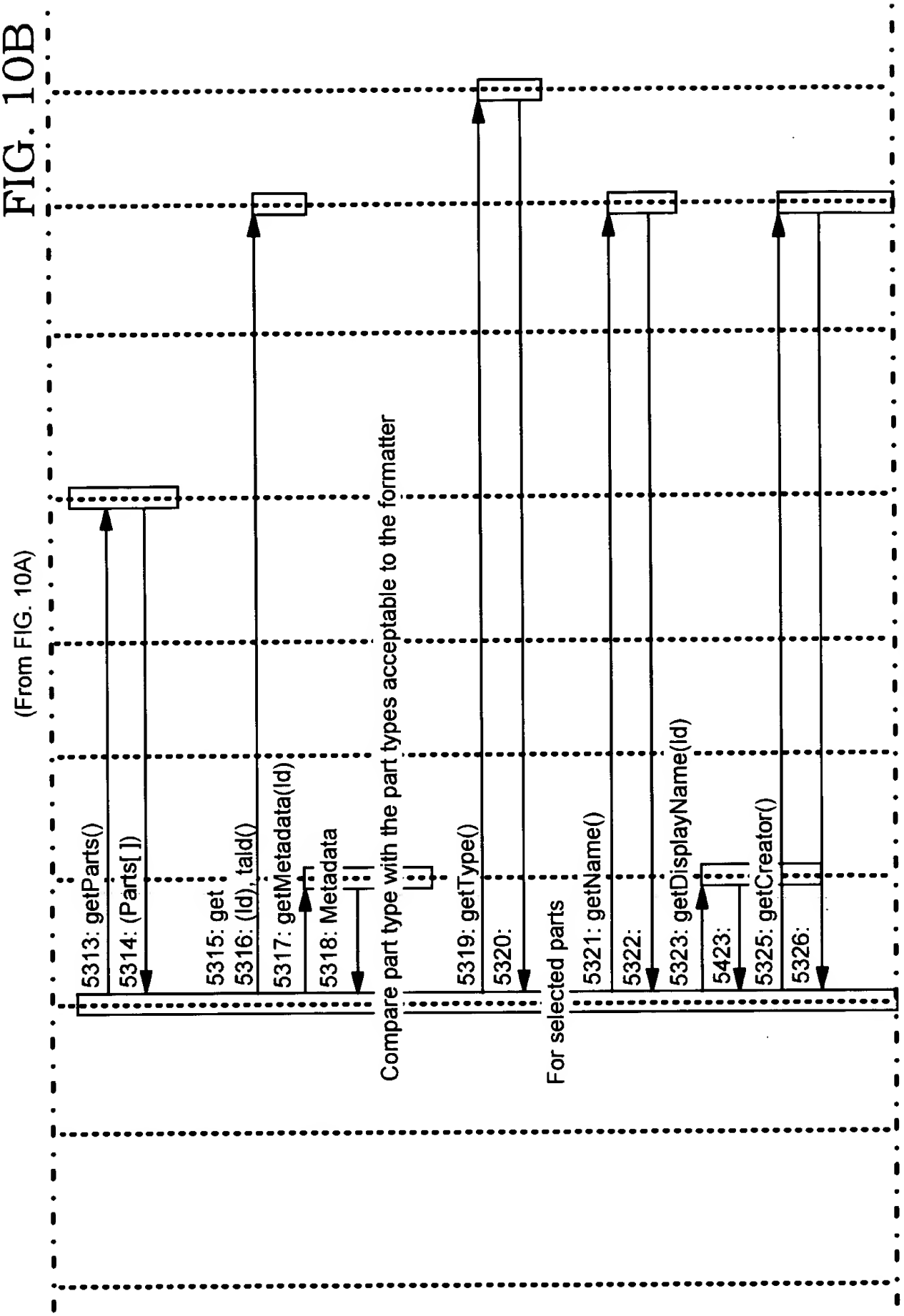

(From FIG. 10A)
5313: getParts()
5314: (Parts[ ])
5315: get
5316: (Id), taId()
5317: getMetadata(Id)
5318: Metadata
Compare part type with the part types acceptable to the formatter
5319: getType()
5320:
For selected parts
5321: getName()
5322:
5323: getDisplayName(Id)
5423:
5325: getCreator()
5326:
(To FIG. 11)

FIG. 11

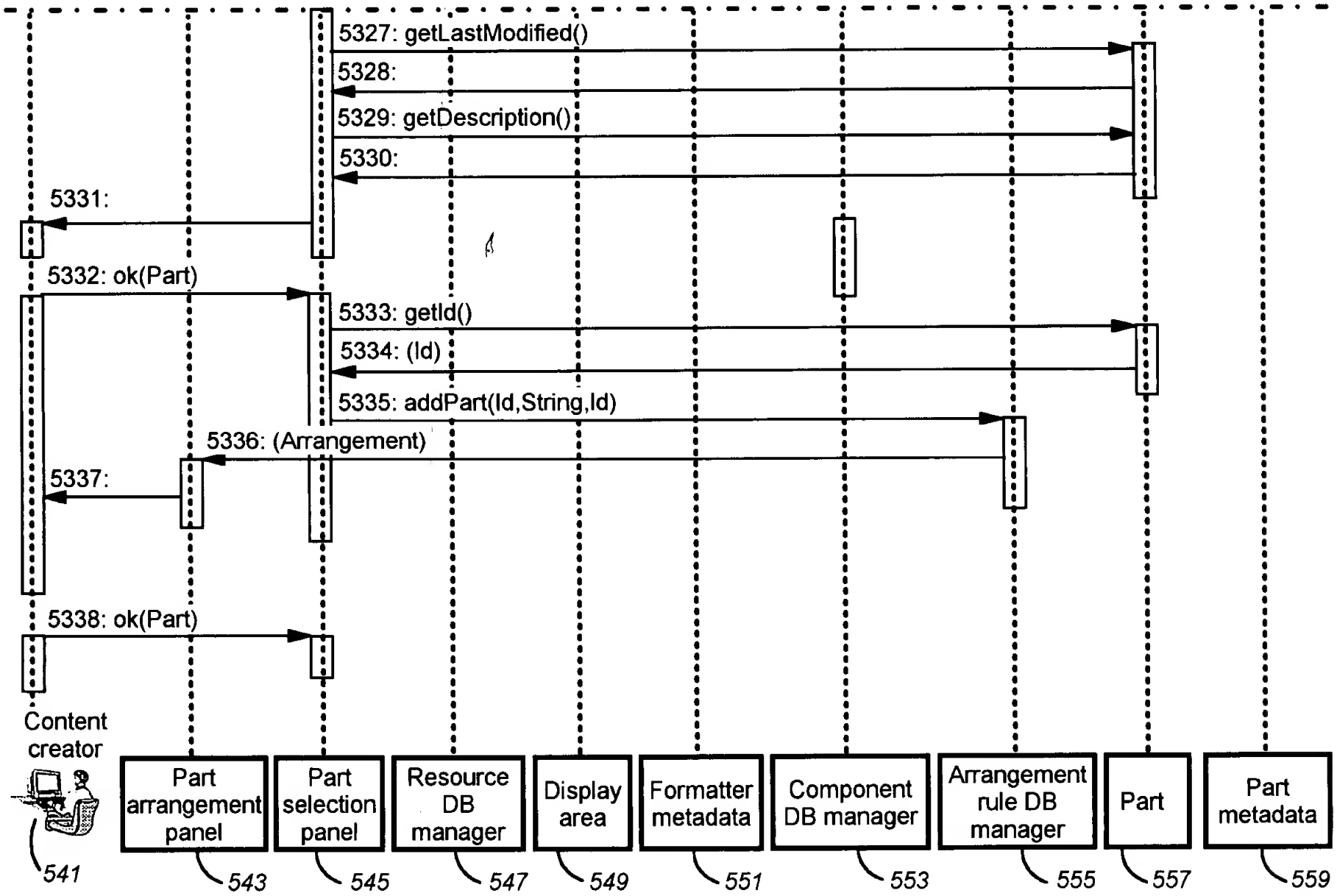

(From FIG. 10B)
5327: getLastModified()
5328:
5329: getDescription()
5330:
5331:
5332: ok(Part)
5333: getId()
5334: (Id)
5335: addPart(Id,String,Id)
5336: (Arrangement)
5337:
5338: ok(Part)
Content creator
541
Part arrangement panel
543
Part selection panel
545
Resource DB manager
547
Display area
549
Formatter metadata
551
Component DB manager
553
Arrangement rule DB manager
555
Part
557
Part metadata
559

FIG. 12

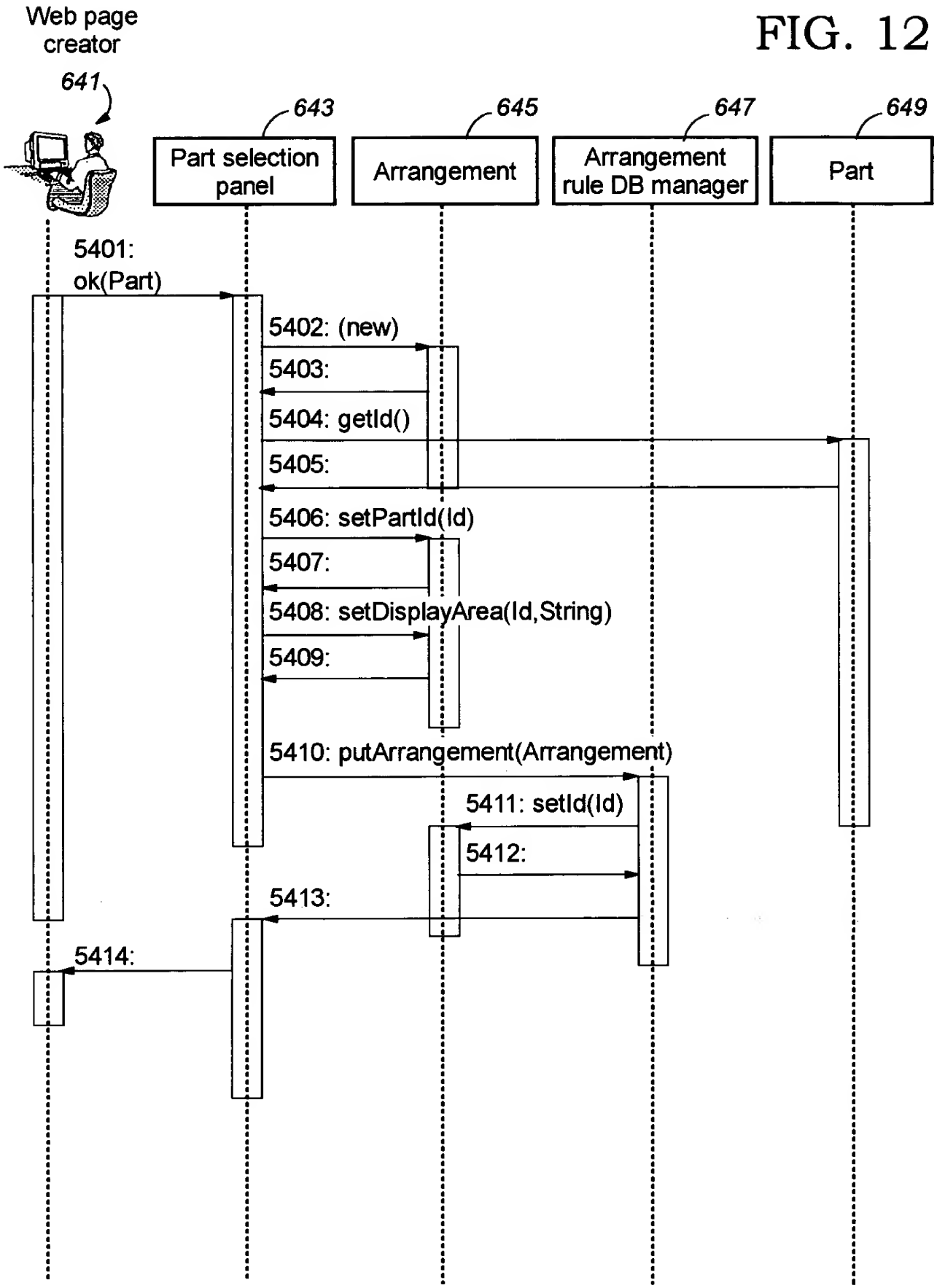

Web page creator
641
643
Part selection panel
645
Arrangement
647
Arrangement rule DB manager
649
Part
5401: ok(Part)
5402: (new)
5403:
5404: getId()
5405:
5406: setPartId(Id)
5407:
5408: setDisplayArea(Id,String)
5409:
5410: putArrangement(Arrangement)
5411: setId(Id)
5412:
5413:
5414:

FIG. 13

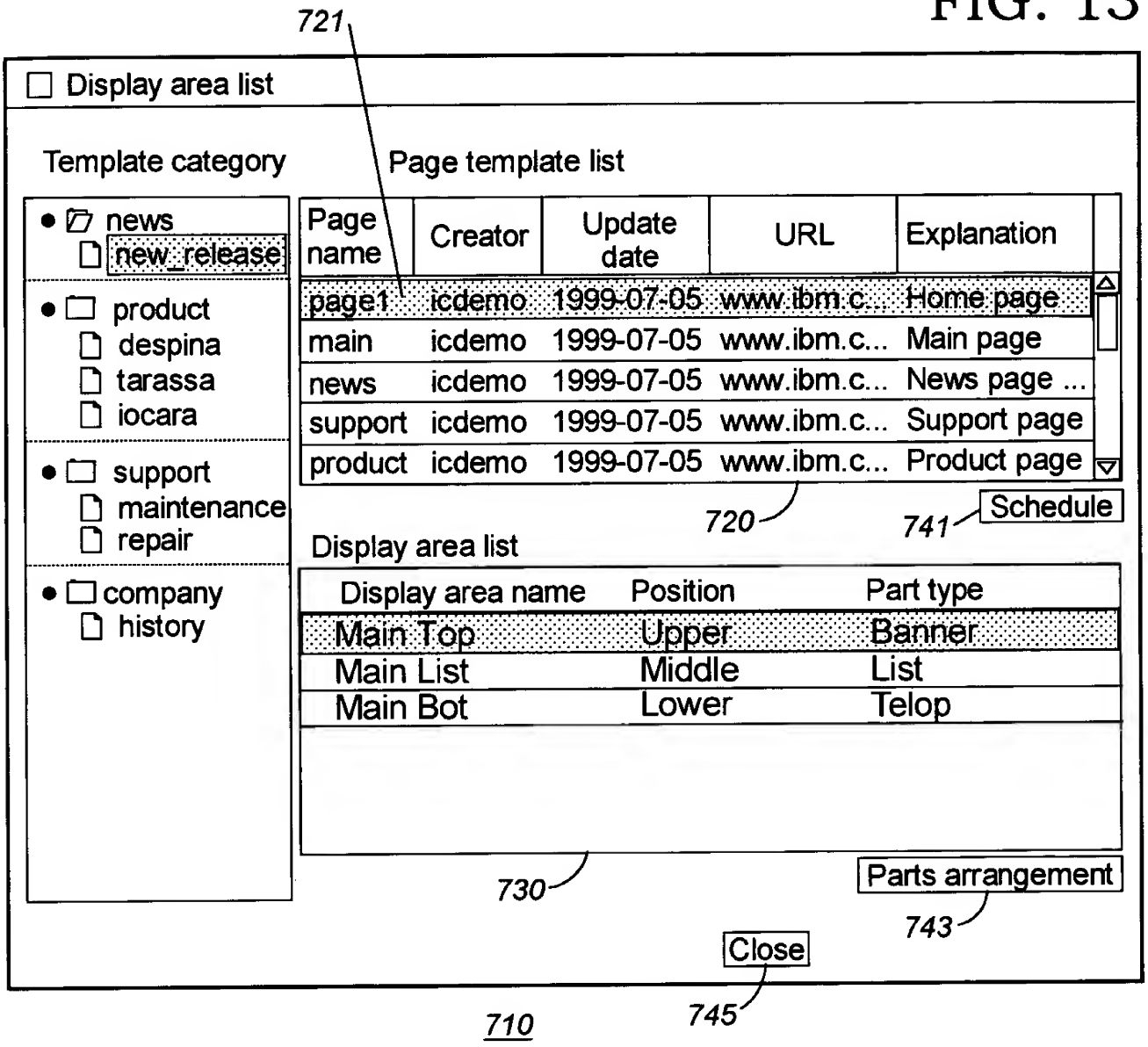

721
Display area list
Template category
Page template list
news
new_release
product
despina
tarassa
iocara
support
maintenance
repair
company
history
Page name
Creator
Update date
URL
Explanation
page1 icdemo 1999-07-05 www.ibm.c... Home page
main icdemo 1999-07-05 www.ibm.c... Main page
news icdemo 1999-07-05 www.ibm.c... News page ...
support icdemo 1999-07-05 www.ibm.c... Support page
product icdemo 1999-07-05 www.ibm.c... Product page
720
741
Schedule
Display area list
Display area name
Position
Part type
Main Top Upper Banner
Main List Middle List
Main Bot Lower Telop
730
Parts arrangement
743
Close
745
710

FIG. 15

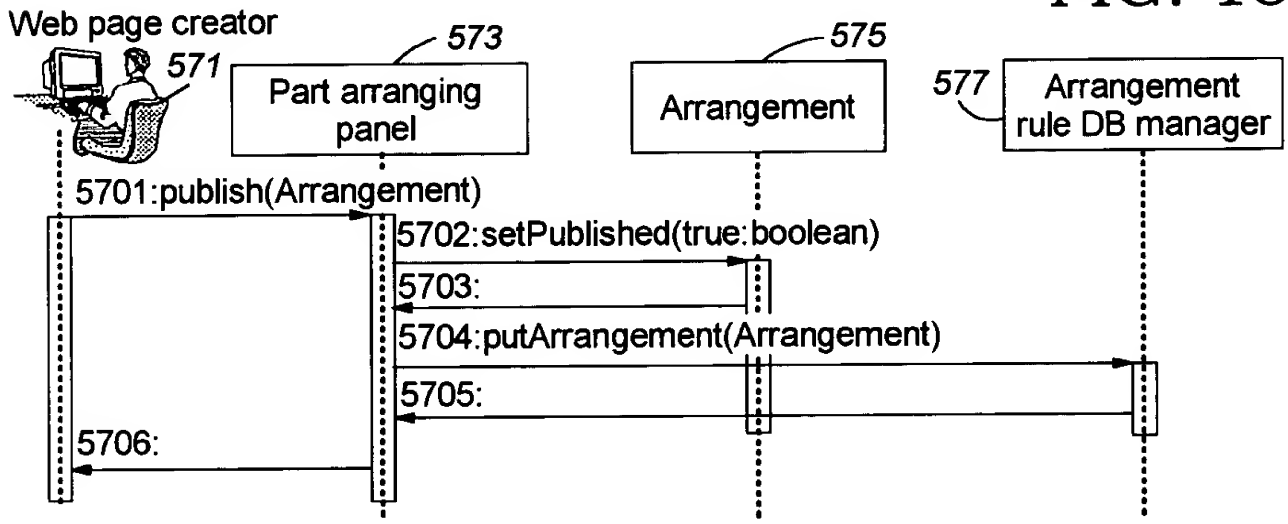

Web page creator
571
573
Part arranging panel
575
Arrangement
577
Arrangement rule DB manager
5701:publish(Arrangement)
5702:setPublished(true:boolean)
5703:
5704:putArrangement(Arrangement)
5705:
5706:

FIG. 14

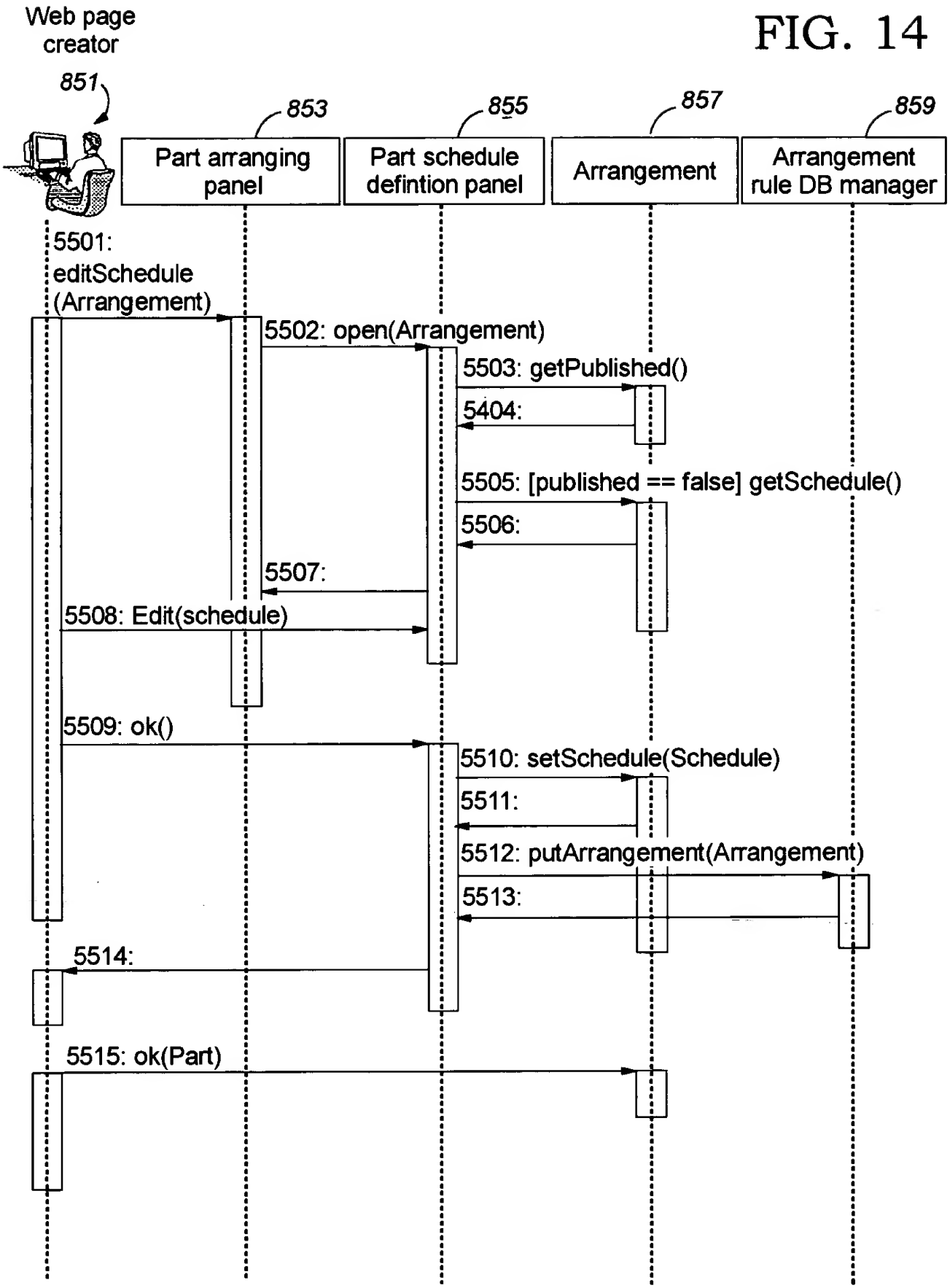

Web page creator
851
853
855
857
859
Part arranging panel
Part schedule defintion panel
Arrangement
Arrangement rule DB manager
5501: editSchedule (Arrangement)
5502: open(Arrangement)
5503: getPublished()
5404:
5505: [published == false] getSchedule()
5506:
5507:
5508: Edit(schedule)
5509: ok()
5510: setSchedule(Schedule)
5511:
5512: putArrangement(Arrangement)
5513:
5514:
5515: ok(Part)

FIG. 16A

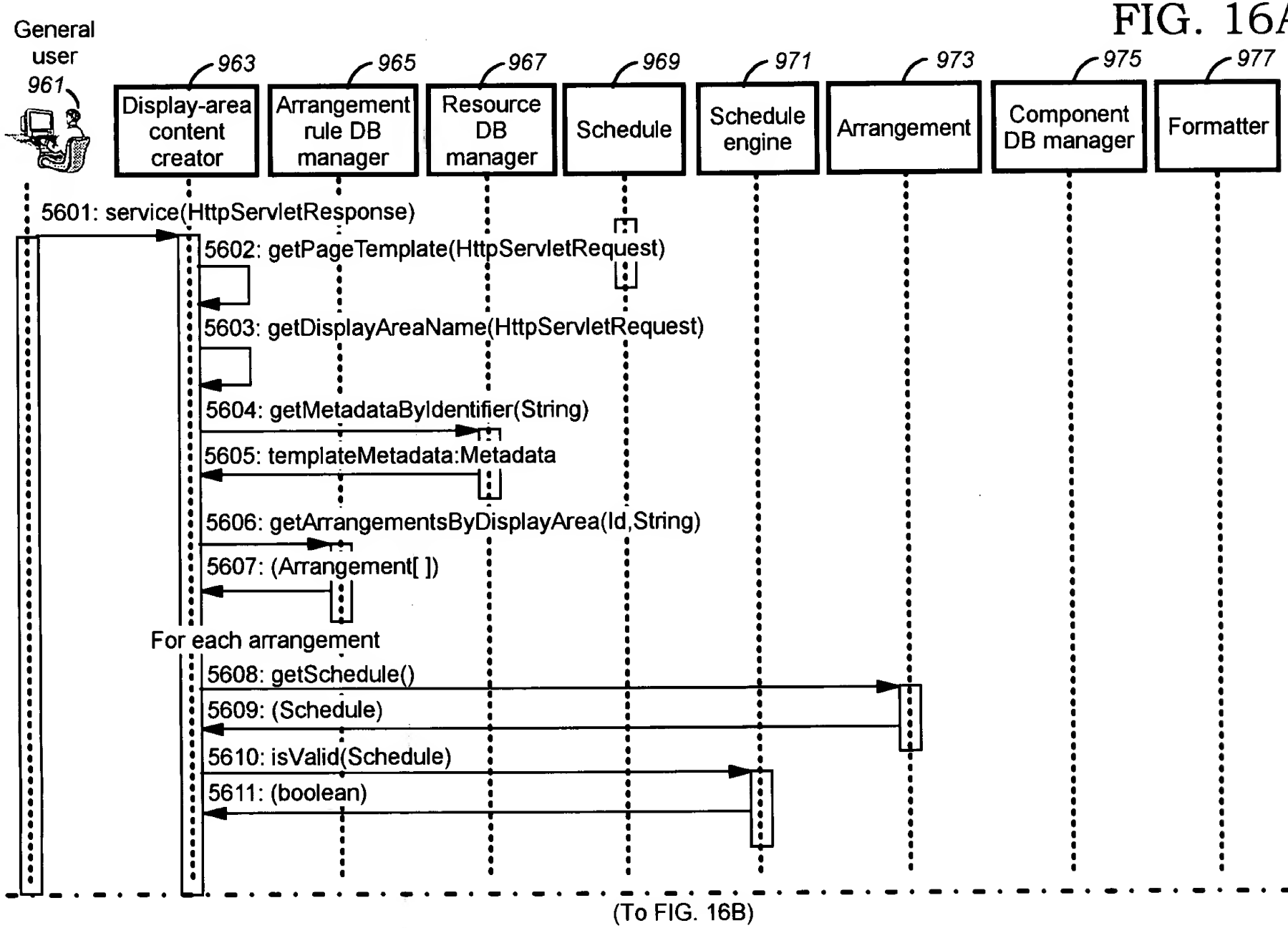
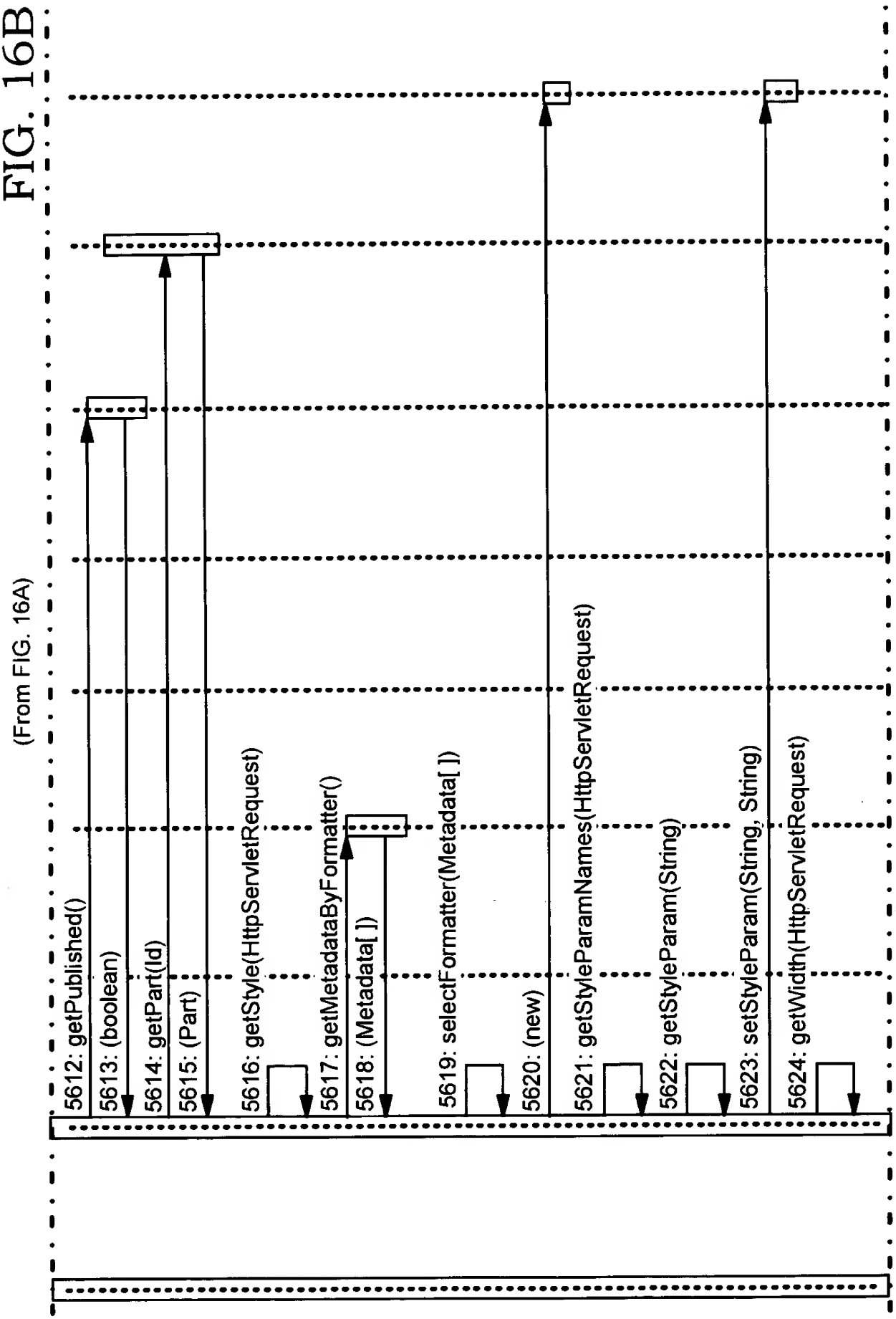

General user
961
963
Display-area content creator
965
Arrangement rule DB manager
967
Resource DB manager
969
Schedule
971
Schedule engine
973
Arrangement
975
Component DB manager
977
Formatter
5601: service(HttpServletResponse)
5602: getPageTemplate(HttpServletRequest)
5603: getDisplayAreaName(HttpServletRequest)
5604: getMetadataByIdentifier(String)
5605: templateMetadata:Metadata
5606: getArrangementsByDisplayArea(Id,String)
5607: (Arrangement[ ])
For each arrangement
5608: getSchedule()
5609: (Schedule)
5610: isValid(Schedule)
5611: (boolean)
(To FIG. 16B)

FIG. 16B


(From FIG. 16A)
5612: getPublished()
5613: (boolean)
5614: getPart(Id)
5615: (Part)
5616: getStyle(HttpServletRequest)
5617: getMetadataByFormatter()
5618: (Metadata[ ])
5619: selectFormatter(Metadata[ ])
5620: (new)
5621: getStyleParamNames(HttpServletRequest)
5622: getStyleParam(String)
5623: setStyleParam(String, String)
5624: getWidth(HttpServletRequest)
(To FIG. 17)

(From FIG. 16B)

FIG. 17

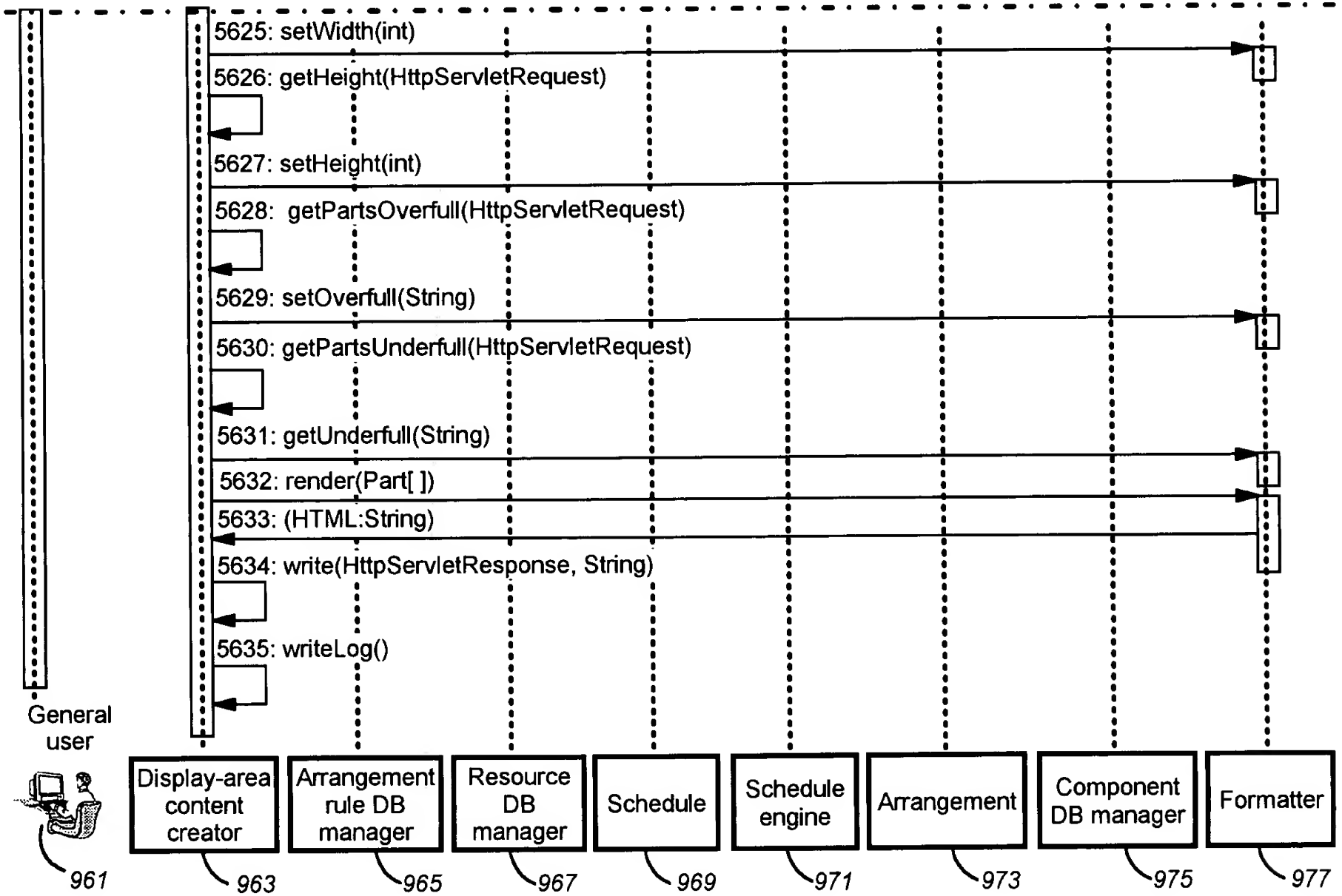


FIG. 18

```
<SERVLET code=icdacrt codebase=servlet>                    701
   <PARAM NAME="name" VALUE="product_ad">
   <PARAM NAME="style" VALUE="bannerFlow">
   <PARAM NAME="direction" VALUE="horizontal">
   <PARAM NAME="bgColor" VALUE="Gray"
   <PARAM NAME="width" VALUE="300">                        703
   <PARAM NAME="height" VALUE="80">
   <PARAM NAME="position" VALUE="top">
   <PARAM NAME="default" VALUE="<IMG SRC='default_img.gif'>">
   <PARAM NAME="partsOverfull" VALUE="random">
   <PARAM NAME="partsUnderfull" VALUE="shrink">
</SERVLET>
```

Example of display area in practical use *700*

FIG. 19

Part ID (704)	Part display area name (705)	Part template ID (706)	START (707)	END (708)	Publish (709)
0001	product_ad	1001	1999-01-01 0:00	1999-07-01 0:00	1
0002	product_ad	1002	1999-07-01 0:00	2000-01-01 0:00	1
0003	product_inf	1003	1999-08-01 4:00	1999-09-20 23:00	0
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.

Arrangement rule DC *710*

FIG. 20

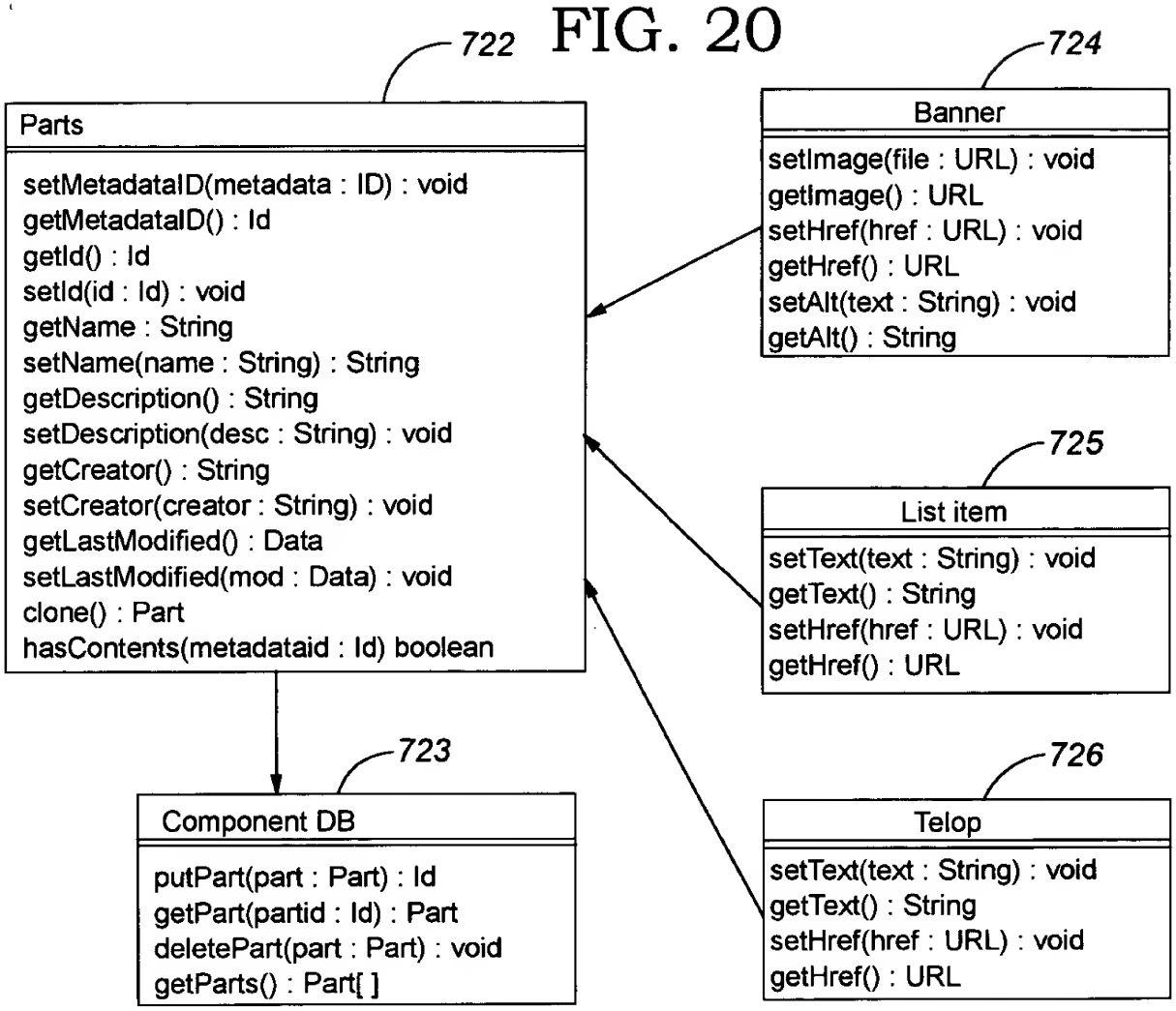

722
Parts
setMetadataID(metadata : ID) : void
getMetadataID() : Id
getId() : Id
setId(id : Id) : void
getName : String
setName(name : String) : String
getDescription() : String
setDescription(desc : String) : void
getCreator() : String
setCreator(creator : String) : void
getLastModified() : Data
setLastModified(mod : Data) : void
clone() : Part
hasContents(metadataid : Id) boolean
724
Banner
setImage(file : URL) : void
getImage() : URL
setHref(href : URL) : void
getHref() : URL
setAlt(text : String) : void
getAlt() : String
725
List item
setText(text : String) : void
getText() : String
setHref(href : URL) : void
getHref() : URL
723
Component DB
putPart(part : Part) : Id
getPart(partid : Id) : Part
deletePart(part : Part) : void
getParts() : Part[ ]
726
Telop
setText(text : String) : void
getText() : String
setHref(href : URL) : void
getHref() : URL

FIG. 21

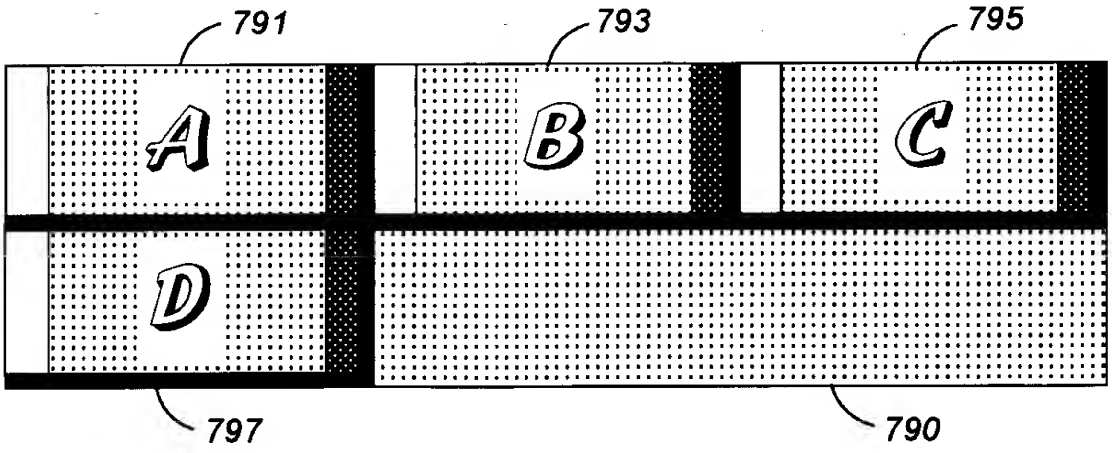

791
793
795
A
B
C
D
797
790

FIG. 22

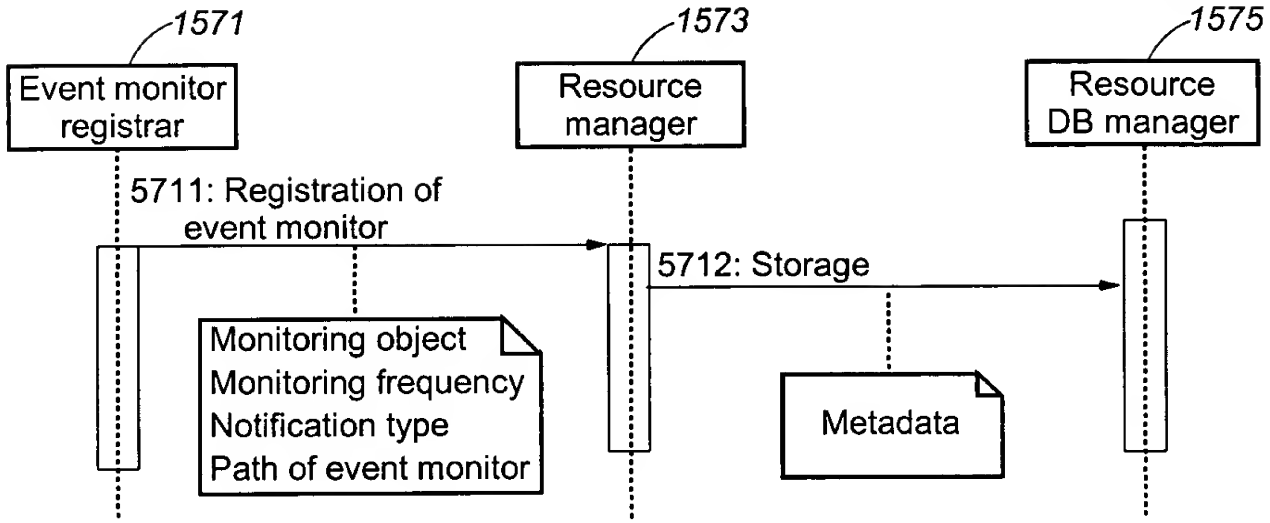

1571
Event monitor registrar
1573
Resource manager
1575
Resource DB manager
5711: Registration of event monitor
Monitoring object
Monitoring frequency
Notification type
Path of event monitor
5712: Storage
Metadata

FIG. 23

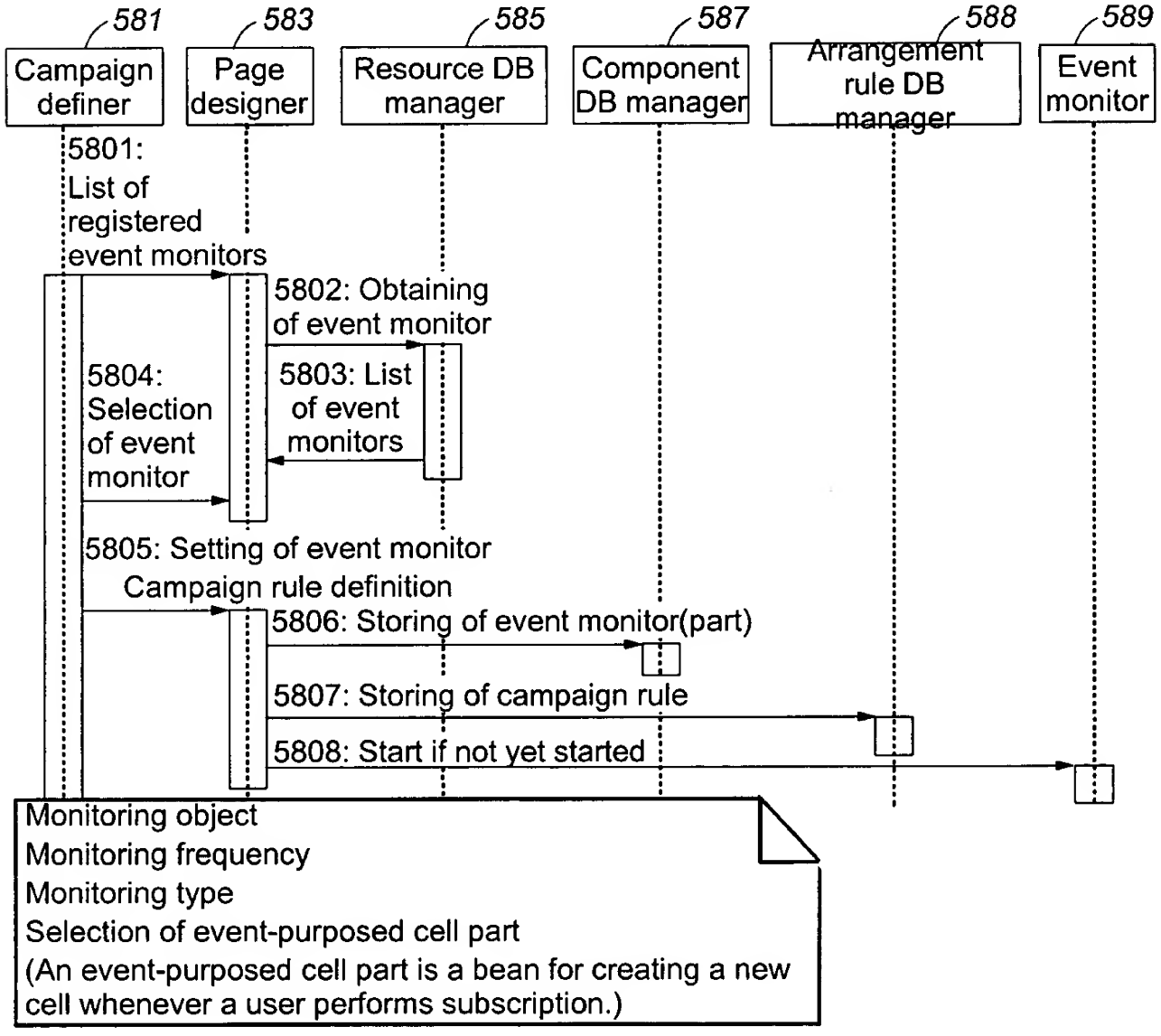

581
Campaign definer
583
Page designer
585
Resource DB manager
587
Component DB manager
588
Arrangement rule DB manager
589
Event monitor
5801: List of registered event monitors
5802: Obtaining of event monitor
5803: List of event monitors
5804: Selection of event monitor
5805: Setting of event monitor
Campaign rule definition
5806: Storing of event monitor(part)
5807: Storing of campaign rule
5808: Start if not yet started
Monitoring object
Monitoring frequency
Monitoring type
Selection of event-purposed cell part
(An event-purposed cell part is a bean for creating a new cell whenever a user performs subscription.)

FIG. 24

1731 Part type	1721 Part ID	1723 Part display area name	1725 Page template ID	1727 START	1728 END	1729 Publish	1733 Event flag	1735 Cell part ID
Banner	0001	product_ad	1001	1999-01-01 0:00	1999-07-01 0:00	1	1	2001
Banner	0002	product_ad	1002	1999-07-01 0:00	2000-01-01 0:00	1	1	2001
Telop	0003	product_inf	1003	1999-08-01 4:00	1999-09-20 23:00	0	0	2001
Event	0004						1	2003
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.

1720
Arrangement rule DB

FIG. 26

1741 User ID	1743 Cookie	1745 E-mail address	1747 User profile

1740
User control DB

FIG. 25

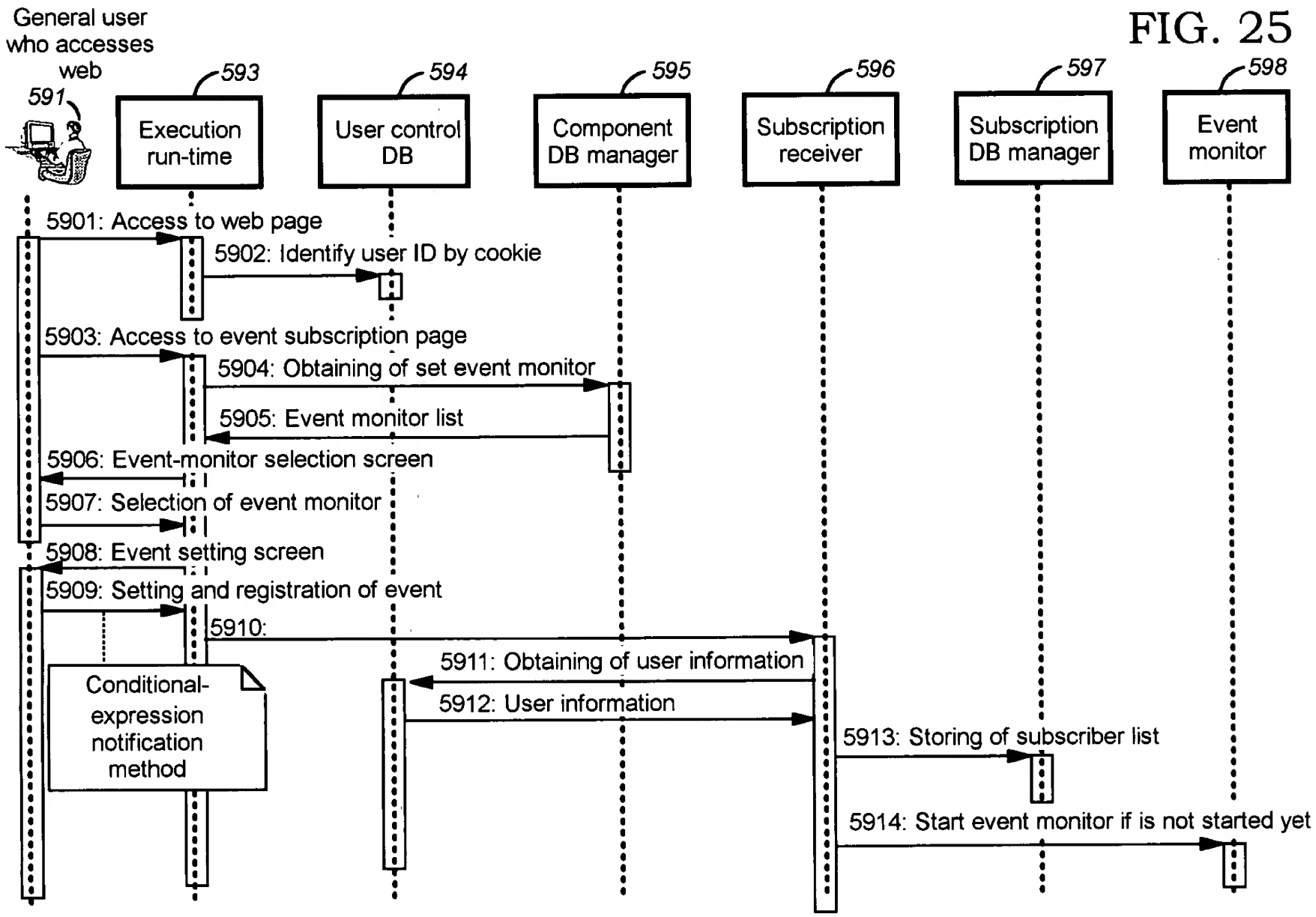
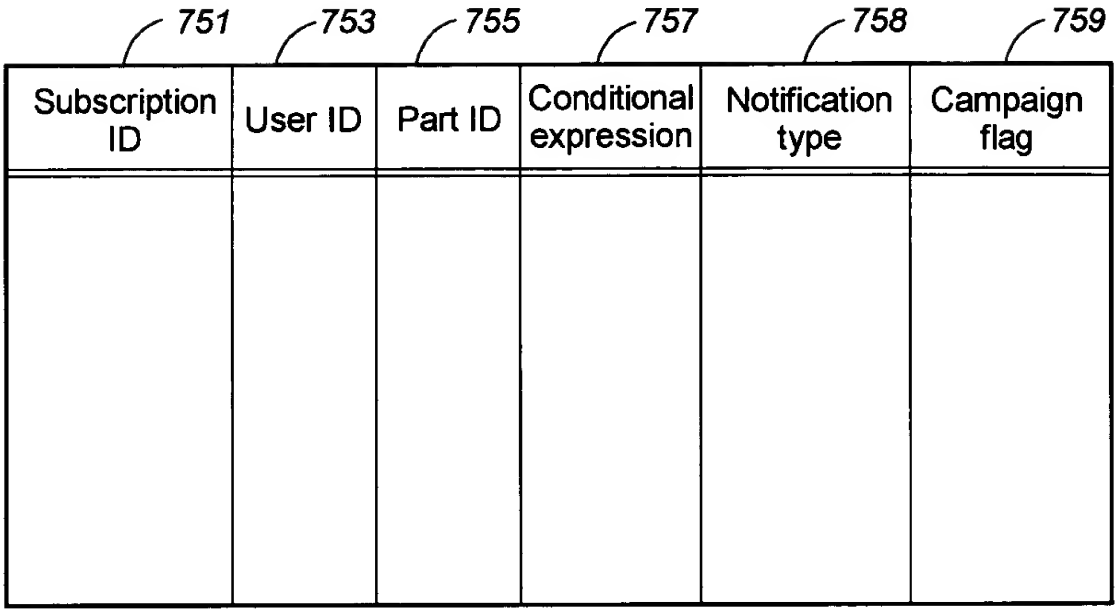

General user who accesses web
591
593
Execution run-time
594
User control DB
595
Component DB manager
596
Subscription receiver
597
Subscription DB manager
598
Event monitor
5901: Access to web page
5902: Identify user ID by cookie
5903: Access to event subscription page
5904: Obtaining of set event monitor
5905: Event monitor list
5906: Event-monitor selection screen
5907: Selection of event monitor
5908: Event setting screen
5909: Setting and registration of event
5910:
Conditional-expression notification method
5911: Obtaining of user information
5912: User information
5913: Storing of subscriber list
5914: Start event monitor if is not started yet

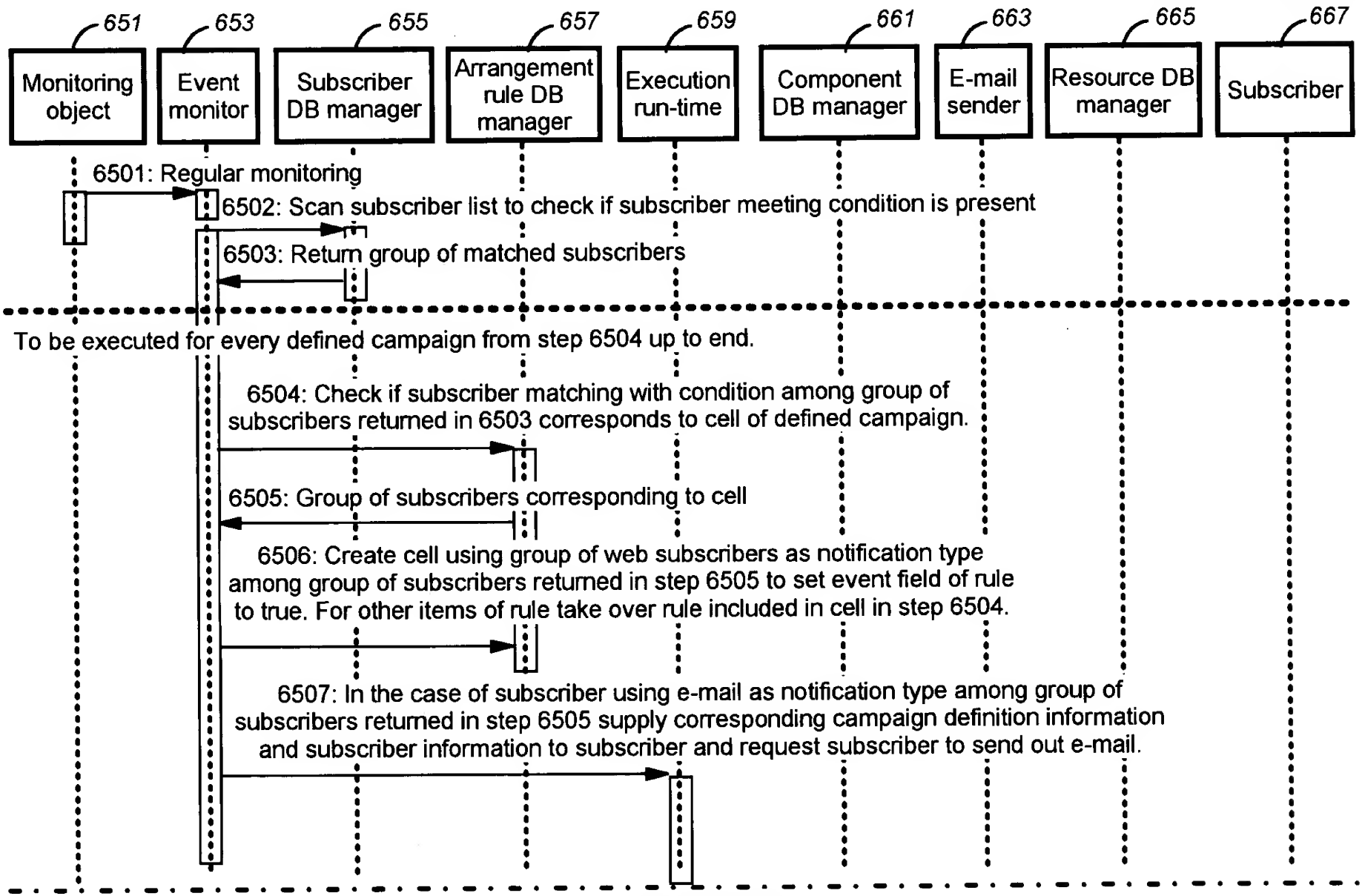
FIG. 27



750
Subscription DB

6,564,209 B1 May 13, 2003

FIG. 28A



(To FIG. 28B)

FIG. 28B

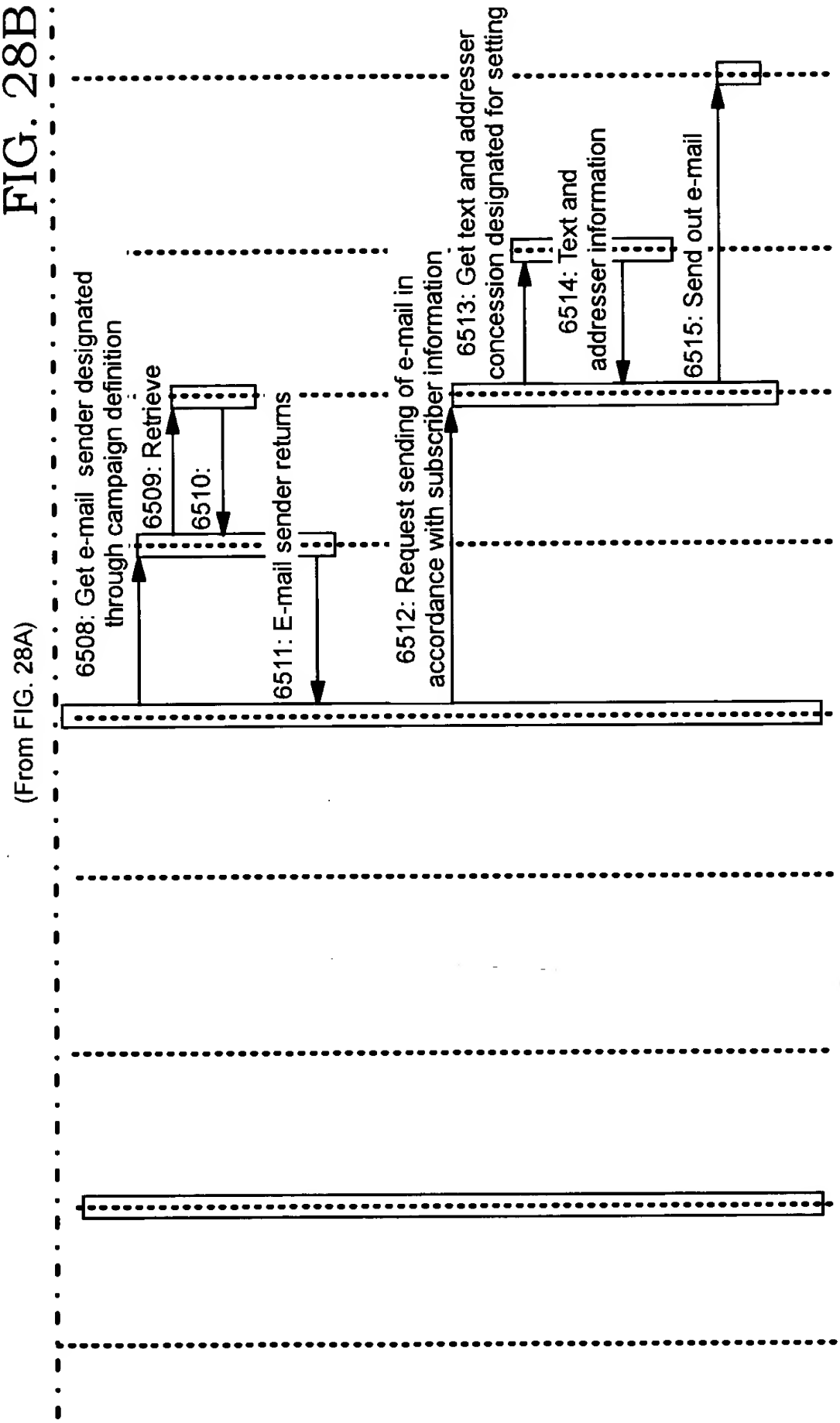

(From FIG. 28A)
6508: Get e-mail sender designated through campaign definition
6509: Retrieve
6510:
6511: E-mail sender returns
6512: Request sending of e-mail in accordance with subscriber information
6513: Get text and addresser concession designated for setting
6514: Text and addresser information
6515: Send out e-mail

FIG. 29
(prior art)

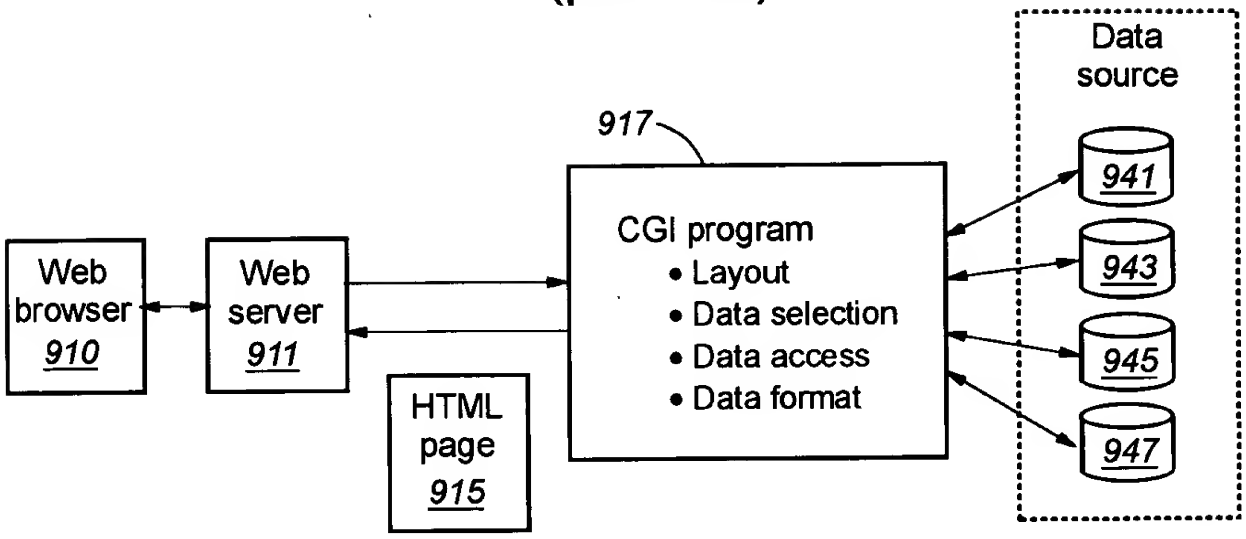
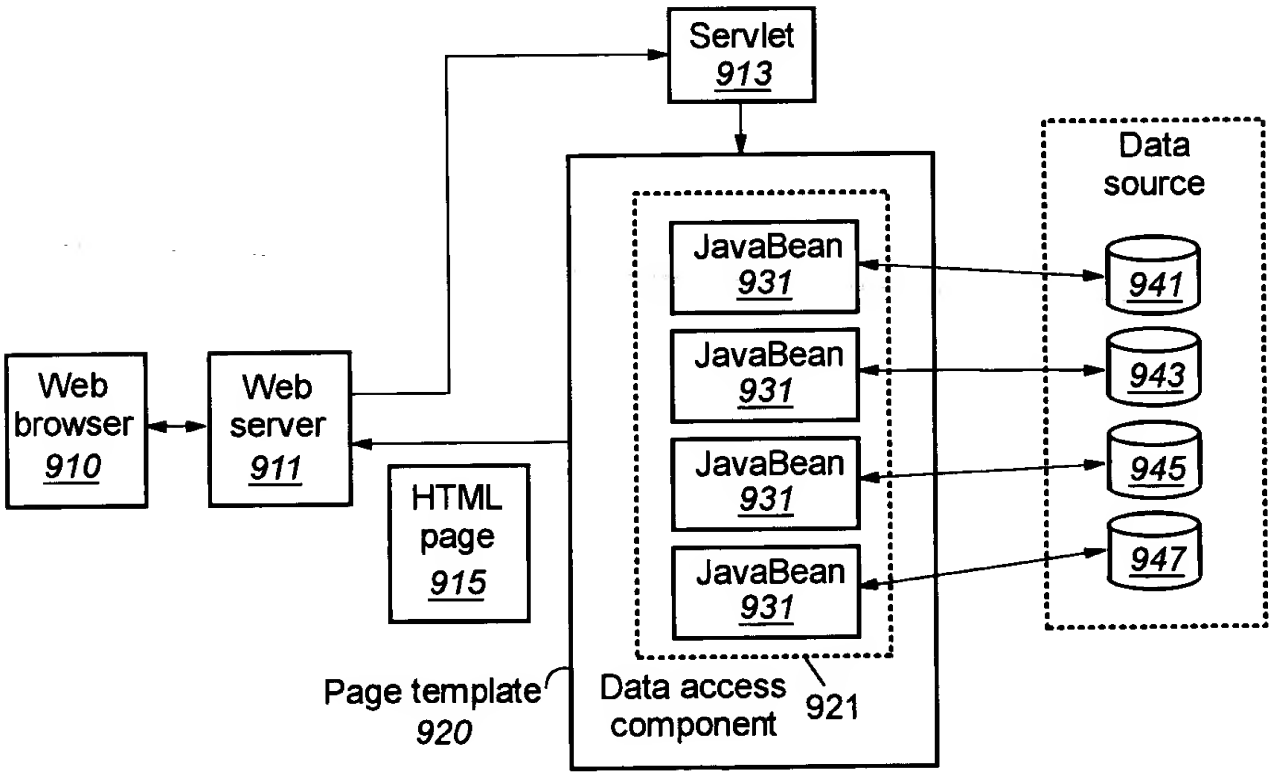



FIG. 30
(prior art)